Pi Presents

Ein Multimedia-Präsentationsprogramm und Animationswerkzeug
für Museen, Besucherzentren und mehr...
...für den Raspberry PI

© Ken Thompson
http://pipresents.wordpress.com

Handbuch
Deutsche Fassung
Peter Vasen (ptv)
http://www.web-echo.de
Dezember 2013

Inhaltsverzeichnis

Copyright-Hinweis

1 Einführung

Pi Presents ist ein Multimedia-Präsentationswerkzeug, das Animationen steuern kann. Das Programm wurde ursprünglich für Museen und Besucherzentren entwickelt. Mittlerweile wird es auch in Krankenhäusern, Shops, Kunstinstallationen und Bibliotheken eingesetzt.

Ich engagiere mich in mehreren gemeinnützigen Organisationen, die unter anderem Museen oder Besucherzentren betreiben. Seit einigen Jahren habe ich nach einer kostengünstigen Möglichkeit gesucht, Audiomaterial in Ausstellungen abzuspielen, so wie Bilder und Videos im Eingangsbereich zu zeigen. Bevor es den Raspberry Pi gab, war der Kauf oder auch der Bau eines einfachen interaktiven Audioplayers teuer. Mit seiner Kombination aus Linux, GPIO und einer leistungsfähigen GPU ist der Raspberry Pi eine ideale Blackbox für Medieninhalte. Es fehlte lediglich noch ein Programm, das auch Nichtprogrammierern ermöglicht, die Power des Raspberry Pi für ihre Zwecke zu nutzen.

Pi Presents unterstützt aktuell fünf Showtypen, die üblicherweise in Museen vorkommen. Jeder Showtyp kann nach den eigenen Bedürfnissen konfiguriert werden. Das Programmwerkzeug enthält einen leicht zu bedienenden Show-Editor, der sowohl auf dem Pi als auch auf einem Windows PC funktioniert.

Das Präsentationswerkzeug bietet:

- Eine Show, in der Sounds, Videos oder Diashows einer Ausstellung mit Buttons, Tastatur oder GPIO-Eingaben animiert oder vorgeführt werden können.
- GPIO-Outputs können während einer Medienpräsentation externe Geräte wie Lampen, Bewegungssimulatoren, Animationstechnik etc. steuern.
- Eine automatische Show für Besucherzentren, die Bilder, Videos, Audiotracks oder Mitteilungen darstellt. Wiederholungen können in einzustellenden Intervallen oder zu bestimmten Tageszeiten laufen.
- Automatische Shows, die vom Besucher unterbrochen werden können und dabei ein Menü mit weiteren Inhalten zeigen.
- Eine Show für Kiosks, auf denen Inhalte durch Druck auf einen von mehreren Knöpfen gestartet werden können.
- Pi Presents kann Shows bereit stellen, wie sie auf Touchscreens in Museen laufen.
- Pi Presents kann „Powerpoint“ ähnliche Multimedia-Präsentationen erstellen, bei denen der Ablauf der Folien manuell durch Buttons oder eine Tastatur gesteuert werden kann.
- Pi Presents eignet sich für ferngesteuertes Abspielen von digitalen Inhalten als Basisshow und ist fähig, über ein Netzwerk Bilder, Videos oder Audiotracks zu laden, um sie in einer in Schleifen laufenden Show einzubauen.
- Pi Presents kann mehrere Shows gleichzeitig abspielen. Dadurch können mehr als ein Exponat oder komplexe interaktive Ausstellungen von einem einzigen Pi gesteuert werden.
- Unterstützung mehrerer Sprachen.

Es sind viele neue Anwendungen für Pi Presents denkbar. Erfahrungen der Programmbenutzer im Echtbetrieb sind für mich von unschätzbarem Wert, weil sie mir beim Feintuning der aktuellen Funktionalität im Kleinen und bei größeren Verbesserungen helfen.

Einmal eingerichtet, ist Pi Presents leicht zu bedienen und erfordert kein Netzwerk:

- Shows können mit einem leicht zu bedienenden Editor auf einem Windows PC, dem Raspberry Pi oder einer Linux-Box vorbereitet werden.
- Die Shows funktionieren auch auf einem Model-A-Raspberry Pi (Hardwareanforderungen beachten).
- Die SD-Card des Pi muss nach der Erstinstallation nicht mehr verändert werden. Alle Medien und Einstellungen können auf einem USB-Stick gespeichert werden. Die Installation neuer Anwendungen ist so einfach, wie der Anschluss eines USB-Sticks.
- Schon beim Pi-Setup kann man für die Videoausgabe zwischen HDMI- und Composite wählen. Weil Pi Presents ohne Tastatur gestartet und heruntergefahren werden kann, ist ein Monitor nicht unbedingt erforderlich.
- Für die Audioausgabe kann man zwischen HDMI und analog wählen und einzelne Spuren links, rechts oder über Stereo-Lautsprecher ausgeben. Mehrere Audiospuren können gleichzeitig auf verschiedenen Lautsprechern ausgegeben werden.

Pi Presents läuft in der Grundeinstellung als tastaturgesteuerte Desktop-Anwendung auf dem Raspberry Pi. Doch mit ein wenig Linux-Zauber, erklärt in diesem Handbuch, kann das Programm auch als Blackbox-Anwendung, gesteuert über GPIO, zum Laufen gebracht werden.
Die eingebauten Blackbox-Funktionen:

- Bildschirmpause und Mauszeiger können deaktiviert werden.
- Vollbildmodus ohne Fensterdekorationen.
- vollständiger Betrieb ohne Tastatur, Maus oder Buttons (wer ängstlich ist, kann einen Button zum Herunterfahren einrichten, um eine eventuelle Korruption der SD-Card zu vermeiden).
- Blackbox-Steuerung durch Buttons, Infrarot-Fernbedienung oder andere digitale Eingangsquellen. Sie können ihre ganz spezielle Steuerung selbst einrichten.
- Maus kompatible Touchscreens werden unterstützt, wenn Treiber für den Raspberry Pi verfügbar sind. Alternativ sind Tastatur und Maus angebracht. In Pi Presents können in Python geschriebene Treiber für andere Arten von Eingabe und Ausgabe, wie RS232 und I2C, geladen werden.
- Automatischer Start, wenn der Raspberry Pi mit Strom versorgt wird.
- sicheres Abschalten ohne Tastatur oder Maus.
- automatisches Herunterfahren des Raspberry Pi zu einer festgelegten Tageszeit.

1.1 Danksagung

Johannes Baiter (jbaiter) für den pyomxplayer auf der Basis von Noah´s pexpect.

Die vielen Aktivisten im Raspberry Pi-Forum und auf Pi-Webseiten, darunter StackOverflow, die mir, ohne es zu wissen, bei der Lösung etlicher technischer Probleme geholfen und Python beigebracht haben.

ActiveState für das freie Python-System für Windows.

Bullets von http://www.enterprise-dashboard.com/tag/red-green-yellow-alert/

Icons von http://www.fatcow.com/free-icons

Raspberry Pi Foundation für die erste erschwingliche Maschine, die Video- und Audio-Tracks abspielt und GPIO anbietet.

Als mich der Raspberry Pi mit Open Source Software zusammenbrachte, war ich erstaunt über die Zeit und den Eifer, womit so viele Menschen Software für andere Menschen programmieren. So fand ich es wichtig, das gleiche mit Pi Presents zu tun. Ich bin froh darüber, denn das Feedback der User und potenziellen Nutzer hat mir sehr viele Ideen für Erweiterungen von Pi Presents eingebracht. Es erübrigt sich, zu sagen, das ich über die viele Arbeit an Pi Presents keine Zeit fand, meine eigenen geplanten Projekte anzugehen. Aber eines Tages, da bin ich ganz sicher, werde ich es tun :-).

2 Installation

Die Installationsanweisungen finden sich in der Datei README.md. Pi Presents ist auf die letzte Raspbian-Version der Foundation aufgesetzt und soll vom LXDE-Desktop installiert und betrieben werden.

3 Beispiele zum Ausprobieren

Download der Beispiele aus dem pipresents-examples Github Repository wie in der README.md Datei beschrieben.

.
Tippen Sie in einem Terminalfenster im Pi Presents-Verzeichnis:

```
python pipresents.py -p pp_mediashow_1p2
```

In einem kleinen Fenster startet eine Show in der Endlosschleife. Das kleine Fenster ist für Entwickler nützlich. Seine Größe kann durch Änderungen im Pythoncode, siehe Kapitel 15 angepasst werden. Die Anzeige kann man mit der zugehörigen Fensterdekoration fast auf Bildschirmgröße bringen. Wenn die Show jedoch ohne Rahmen und komplett auf dem Vollbildschirm erscheinen soll, ist die Kommando-Option -f erforderlich.

```
Python pipresents.py -p pp_mediashow_1p2 -f
```

Das Mediashow-Beispiel und alle anderen Beispiele sind für eine Bildschirmauflösung von 1920x1080 geeignet. Bei Bildschirmen mit geringerer Auflösung liegen einige Textteile und Teile größerer Bilder außerhalb des Bildschirmrandes. Pi Presents passt die Auflösung nicht automatisch an die Bildschirmgröße an. Die Textgröße kann im Pi Presents Editor angepasst werden, während Bilder außerhalb von Pi Presents auf die richtige Bildschirmauflösung gebracht werden müssen.

Im Pi Presents-Repository befinden sich Beispiele für Anwendungen in Pi Presents

- pp_mediashow_1p2
  Das ist das oben heruntergeladene Profil. Mediashow startet sofort, fährt automatisch fort und beginnt nach 200 Sekunden wieder von vorn. Mit den Pfeiltasten für Auf und Ab können Sie den laufenden Showteil überspringen.
- pp_menu_1p2
  Verwenden Sie die Pfeiltasten Auf und Ab, um durch das Menü zu wandern. Die Eingabetaste startet einen Menüpunkt, Escape beendet ihn.
- pp_exhibit_1p2
  Dieses Profil zeigt, wie eine Mediashow (nicht nur) mit der Infrarotfernbedienung (PIR) gesteuert werden kann. Beim Start ist der Bildschirm frei. Durch das Auslösen der PIR (öffnet einen Kontakt, der mit 0 Volt an P1-11 angelegt ist) oder durch Druck auf einen Button (schließt einen Kontakt mit 0 Volt, der an P1-18 angelegt ist) wird eine Mediashow einmalig gestartet. Danach wartet Pi Presents auf einen neuen Impuls.
  Das Profil benutzt eine gpio.cfg, in der die Pins P1-11 und P1-18 den symbolischen Namen PIR tragen. Dieser Name ist mit dem Startbefehl für die Mediashow verknüpft.
  Das Beispiel benutzt die Eingabetaste als Steuertaste zusätzlich zu den GPIO-Pin-Inputs. So kann man den Effekt sehen, ohne die GPIO-Pins zu benutzen. Wenn der Start mit der Eingabetaste ausgeschaltet werden soll, muss Disable Controls auf Yes umgeschaltet werden.
  ACHTUNG: Vor der Benutzung der GPIO-Pins Kapitel 9.4 lesen.
- pp_interactive_1p2
  In diesem Beispiel sind Mediashow und Menü kombiniert. Die Mediashow läuft kontinuierlich, aber jeder Beitrag zeigt den Hinweis, dass die Eingabetaste ein Menü öffnet. Durch Druck auf Escape geht es wieder zurück zur Mediashow.
- pp_presentation_1p2
  Die MediaShow ist als Präsentation gestaltet. Sie wird mit der Eingabetaste und der Ab-Pfeiltaste gestartet und kann mit den Auf- und Ab-Pfeiltasten durchsucht werden. Escape stoppt ein laufendes Video und wartet auf einen neuen Befehl.
- pp_liveshow_1p2
  Platzieren Sie bei laufender Liveshow Videos, Audiodateien, Internetseiten oder Bilder im Verzeichnis /home/pi/pp_home/pp_live_tracks und erleben Sie, wie sie wundersamerweise erscheinen.
- pp_radiobuttonshow_1p2
  Radibuttonshow, gesteuert von der Tastatur oder GPIO-Pins. Über die Links im Radiobuttonshow-Profil können mit der Tastatur und Buttons Tracks ausgewählt werden. Zusätzlich enthält das Profil eine gpio.cfg Datei, in der vier GPIO-Pins mit Eingängen verbunden werden können. Ein weiterer Pin ist mit der internen Stop-Funktion verbunden.
- pp_hyperlinkshow_1p2
  Dieses Beispiel zeigt die Möglichkeiten der Hyperlinkshow. Die Show wird über Buttons auf der Bildschirmoberfläche mit Mausklick gesteuert. Eigentlich ist ein Touchpad ideal,

allerdings besitze ich so ein Eingabegerät nicht und konnte es deshalb nicht testen. Die Datei screen.cfg definiert die Buttons und die symbolischen Namen ihres Eingabebefehls. Die Show kann über GPIO-Pins gesteuert werden, wenn sie in der gpio.cfg Datei mit den symbolischen Namen verknüpft sind.

- pp_audio_1p2
  Diese mediashow demonstriert die Fähigkeiten des audioplayers. Um sie vollständig nutzen zu können, sind Lautsprecher am HDMI- und Analog-Port erforderlich. Die Show spielt Audiotracks auf verschiedenen Lautsprechern ab. Sie zeigt überdies die Funktion der Laufzeitkontrollen, mit denen die Lautstärken per Tastatur oder GPIO-Pins geregelt werden können. Dafür gibt es im Profil die Dateien controls.cfg und gpio.cfg.
- pp_web_1p2
  Eine Demonstration der Webbrowser-Fähigkeiten, für die eine Internetverbindung vorhanden sein sollte. Das Profil enthält die Dateien keys.cfg und controls.cfg, die es ermöglichen, den Zoom des Browsers mit den l- und s-Tasten zu steuern. Dafür ist es erforderlich, zuerst auf das Hintergrundbild zu klicken.
- pp_concurrent_1p2
  Diese Anwendung zeigt, wie zwei Shows gleichzeitig abgespielt werden können. Zwei Shows laufen simultan. Die Show mit den Bildern wird über die Tastatur gesteuert, während die konkurrierende Mediashow mit Audiotracks bei abgeschalteter Steuerung kontinuierlich läuft.
- pp_subshow
  Zeigt, wie man größere Shows mit Subshows aufteilen und damit übersichtlicher machen kann.
- pp_timeofday_1p2
  Steuert zwei Mediashows zu unterschiedlichen Tageszeiten. Um diese Demonstration zu ermöglichen habe ich die Tageszeiten durch Verzögerungen (+xx) ersetzt. Sie können mit Lokalzeiten ausgetauscht werden, z.B. 10.30.
- pp_plugin_1p2
  Zeigt die Arbeitsweise von Plugins. Einige Beispiele benötigen einen Internetanschluss. Bevor die Liveshow gestartet wird, müssen die cfg. Dateien vom /media Verzeichnis der Beispiele in das Verzeichnis /live_tracks kopiert werden.
- pp_showcontrol_1p2
  Zeigt, wie Show Control konkurrierende Shows nacheinander startet und stoppt.
- pp_animate_1p2
  Animationen werden über den Pin P1-11 mit On am Beginn und mit Off am Ende des Tracks gesteuert. Für die Animationssteuerung gibt es eigens eine gpio.cfg Datei.
- ACHTUNG: Stellen Sie zuvor sicher, dass P1-11 ungefährdet als Ausgang genutzt werden kann.
- pp_shutdown_1p2
  Zeigt, wie Show Control für das Herunterfahren des Pi genutzt wird. Eine entsprechende Meldung wird angezeigt. Wenn die Ab-Pfeiltaste gedrückt wird, fährt ein so gestarteter Track den Pi sofort herunter.

Sie können die Beispiele mit dem Bash-Script examples.sh ausführen. Lediglich /examples.sh in einem geöffneten Terminalfenster im Verzeichnis /pipresents eingeben. Die Datei examples.sh muss ausführbar gemacht werden.

Alle Beispiele verwenden eine Auswahl von Medien, die in /home/pi/pp_home/media gespeichert sind. Die Profile befinden sich in /home/pi/pp_home/pp_profiles. In jedem Profil beschreibt die Datei pp_showlist.json das „Look and Feel" jeder Show, andere Dateien regeln die Anzeige der benutzten Mediadateien. Alle Dateien können mit einem Texteditor angezeigt werden, aber es ist viel besser, sie in PiPresents eigenem Editor zu bearbeiten, wie in Kapitel 4 beschrieben.

Die Tastatur-Befehle:

- Auf- und Ab- Pfeiltasten führen durch das Menü
- Auf- und Ab-Pfeiltasten führen durch eine Mediashow. Mit ihnen kann man auch zum nächsten oder einem früheren Track einer automatischen Show springen. Die Bewegung beginnt am Start und Ende der Show neu.
- Eingabe startet eine Präsentation oder den ausgewählten Menüeintrag.
- Escape stoppt einen laufenden Beitrag und kehrt zur Show zurück. In einer Show führt Escape zur führenden Show zurück.
- Leertaste - in einem Bild, einem Video oder einer Audio-Spur ermöglichen 'p' oder die Leertaste eine Pause und dann die Fortsetzung.
- Strg-Pause schließt Pi Presents.

3.1 Pi Presents beenden und den Pi herunterfahren

Standardmäßig wird Pi Presents mit Strg-Pause oder Click auf das Schließen-Symbol beendet. 5 Sekunden Druck auf den Button am GPIO-Pin, der mit der Shutdown-Funktion verbunden ist, fährt den Pi herunter Kapitel 9.7.

Andere Wege, Pi Presents zu beenden:

- Einige Tastaturen haben keine Pause-Taste; dann kann eine andere Taste für das Beenden des Pi konfiguriert werden, indem sie in der keys.cfg mit pp_exit verbunden wird, wie in Kapitel 10.2 beschrieben.
- Eine andere Taste, einen GPIO-Pin, oder einen Klickbereich mit dem symbolischen Namen pp_exit verknüpfen, wie in Kapitel 10.2 beschrieben.
- Den exit-Befehl der Show Control anwenden, um Pi Presents innerhalb einer Show zu beenden (siehe Kapitel 6.4.3).

Andere Möglichkeiten, aus Pi Presents heraus den Pi herunterzufahren:

- Standardmäßig wird ein Button 5 Sekunden lang gedrückt, der mit dem GPIO-Pin, unter dem symbolischen Namen pp_shutdown verbunden ist (siehe Kapitel 10.2). Es kann nur ein Pin mit pp_shutdown verknüpft sein.
- Verknüpfen Sie eine beliebige Taste, einen GPIO-Pin oder einen Klickbereich mit dem symbolischen Namen pp_shutdownnow, wie in Kapitel 10.2 beschrieben. Eine Eingabe unter diesem symbolischen Namen lässt den Raspberry Pi sofort herunterfahren.

- Um den Pi sofort innerhalb einer Show herunterzufahren, benutzen Sie das Show Control Kommando „shutdownnow“ (siehe Kapitel 6.4.3)

4 Der Pi Presents Profil-Editor

Mit dem pp_editor.py können Profile erstellt und bearbeitet werden:

```
python pp_editor.py
```

Wenn Sie den Editor benutzen, ist es erforderlich, Show-Referenzen, Track-Referenzen und Dateinamen einzurichten. Es ist ratsam, Namen nicht mit pp- oder pp_zu beginnen, um Konflikte mit Namen zu vermeiden, die Pi Presents benutzt.

4.1 Benutzung des Editors

4.1.1 Die Programmfenster

Wählen Sie eines der Beispielprofile mit dem Menübefehl Profile > Open.
Achtung: In einem Linuxsystem zuerst das gewünschte Verzeichnis öffnen und dann OK klicken.

- Das obere linke Fenster zeigt die im gewählten Profil verfügbaren Shows
- Das untere linke Fenster zeigt die Medienlisten dieses Profils. Mit einem Klick wird eine Medienliste markiert.
- Das rechte Fenster zeigt die Tracks der ausgewählten Medienliste.

Die ausgewählten Einträge werden rot dargestellt. Klicken Sie auf die Schaltfläche 'Edit Show' neben dem linken Fenster, um die ausgewählte Show zu bearbeiten. Die ausgewählten Tracks werden nach Klick auf die Edit-Schaltfläche neben dem rechten Fenster bearbeitet.

Änderungen werden mit Klick auf OK gespeichert, mit Cancel bricht der Vorgang ab.

Um Auswirkungen der Änderungen gleich sehen zu können, sollten Sie pipresents.py und pp_editor.py nebeneinander in zwei Terminalfenstern öffnen. Pipresents.py zeigt die Änderungen, allerdings erst, wenn es neu gestartet wurde.
Wenn Sie die Beispiele bearbeiten wollen, sollten Sie zuvor Kopien der Originale anfertigen, indem Sie mit dem Dateimanager aus dem Verzeichnis /home/pi/pp_home/pp_profiles den gewünschten Ordner kopieren.

4.1.2 Profil-Menü-Funktionen

Profile > Open öffnet ein Verzeichnisfenster, in dem Profile zur Bearbeitung ausgewählt werden können. Es handelt sich dabei um Profile im home-Verzeichnis, das im Menü Options > Edit Menu eingetragen ist.

Profile > Validate überprüft das Profil. Schrift- und Farbfelder werden nicht überprüft. Wenn die .json-Dateien lediglich mit einem Texteditor bearbeitet werden, geschieht das auf eigenes Risiko!

Profile > New From Template zeigt eine Liste von Vorlagen für alle Beispielshows. Die Vorlagen verfügen über Beispiel-Tracks die auch unbearbeitet laufen. Einigen fehlen allerdings die Kofigurationsdateien.

4.1.3 Show Menu

Show>Add - Fügt eine neue Show und eine Medialiste mit dem gleichen Namen hinzu. Der Showtyp einer bereits vorhandenen Show kann nicht geändert werden.

Show>Copy To – Kopiert die Show und erstellt eine leere Medialiste für diese Show.

Show>Edit – Dupliziert den Edit Show Button.

Show>Delete – Löscht die Show aus dem Profil, rückgängig nicht möglich.

4.1.4 Medialist Menu

Medialist>Add – Fügt eine neue Medialiste hinzu. Die Endung .json wird angefügt, falls nicht angegeben.

Medialist>Delete – Löscht die Medialiste aus dem Profil, rückgängig nicht möglich.

4.1.5 Track Menu

Track> Add Kapitel 6.2.1 from File
Fügt einen oder mehrere Trackeinträge mit Mediadateien an das Ende der Medialiste an. Das können Bilder, Videos oder Audiotracks sein. Der Editor erkennt automatisch den Speicherort und den Tracktyp. Sollte eines von beiden nicht erkannt werden oder die Dateiendung ist verschwunden, so fügen Sie mit Track>New einen leeren Track des gewünschten Typs hinzu.

Eine Liste der unterstützten Dateierweiterungen findet sich in den ersten Zeilen der pp_definitions.py Quelldatei. Bitte einen Fehlerreport senden, wenn die Aufstellung unvollständig ist.

Track>Add from Dir
Wie Add from File, fügt jedoch alle erlaubten Dateien aus dem Verzeichnis der Medialiste hinzu. Eine Liste der unterstützten Dateierweiterungen für die Auswahl von Tracks und Tracktypen findet sich in den ersten Zeilen der pp_definitions.py Quelldatei. Bitte einen Fehlerreport senden, wenn die Aufstellung unvollständig ist.

Track>New
Fügt einen leeren Track des gewählten Typs am Ende der Medialiste hinzu.

Track>Edit
Dupliziert den Edit Button

Track>Delete
Löscht den Track aus der Medialiste, rückgängig nicht möglich.

4.1.6 Tools Menu

Tools>Update All
Versionsupdate für alle Profile im aktuellen Daten-Heimverzeichnis (siehe Kapitel 13).

4.1.7 Options Menu

Options>Edit
Bearbeiten der Editoreinstellungen

- Pi Presents Heimatverzeichnis
  Diese wichtige Einstellung muss das Verzeichnis nennen, in welchem alle Profilverzeichnisse und die zugehörigen Medien aufbewahrt sind. Standard ist /home/pi/pp_home. Wenn ein alternativer Speicherort gewählt werden soll, ist sicherzustellen, dass das Verzeichnis /pp_profiles in dem nun gewählten pp_home erstellt wird.
- Einstieg Medienverzeichnis
  Nur eine Hilfe, für den Suchanfang für 'Add Track' und 'Add from Dir'.

4.1.8 Editor Befehlszeilen-Optionen

-d --debug	Laufzeitfehler lösen Alarm aus und werden im Terminalfenster angezeigt, wenn es geöffnet ist. Wenn debugging eingeschaltet ist, wird ein Verlaufsprotokoll der Editor-Operationen im Terminalfenster angezeigt. Fehler und Verlauf sind zudem in der Datei /pipresents/pp_log.log aufgezeichnet.
--forceupdate	Update eines Profils, auch wenn die Ausgaben von Editor und Profil übereinstimmen (nur für Betatester).

4.2 Profile erstellen

Die Anwendungsdaten liegen im Verzeichnis /pp_home. Der Ordner /pp_home muss ein Verzeichnis /pp_profiles enthalten, in dem die Profile gespeichert sind.
Ein Profil ist ein Verzeichnis, dass die Informationen enthält, die man benötigt, um eine einzelne Anwendung in Pi Presents zu konfigurieren. Es muss eine pp_showlist.json Datei und eine oder mehrere medialist (.json) Dateien enthalten. Die showlist.json enthält mehrere Bereiche. Einen 'start' show Bereich und mehrere Bereiche, in denen die vom Benutzer erstellten Shows definiert werden. Jeder Bereich in der pp_showlist.json bestimmt das 'Look and Feel' einer Show und regelt die Verknüpfungen. Medialist (.json) Dateien bestimmen den Inhalt jeder Show und einige Gestaltungsinformationen für die Tracks.

Aus Gründen der Portabilität sind alle Medien am besten im Verzeichnis /pp_home aufgehoben, das ist aber nicht zwingend erforderlich. Medien können, wenn es erforderlich ist, überall in einem Verzeichnis /pp_home in Unterverzeichnissen gespeichert werden.

Der Standardspeicherort für /pp_home ist /home/pi; das muss durch Befehlszeilenoptionen überschrieben werden, wenn ein USB-Stick benutzt wird. Es gibt auch ein /pp_home Verzeichnis

im /pipresents Verzeichnis. Das sollte aber nicht benutzt werden, weil es die Vorlagen für den Editor so wie Standard Konfigurationsdateien enthält.

4.2.1 Profile portieren

Ich wollte ein System entwickeln, in dem Profile und ihre Medien zwischen verschiedenen Laufwerken hin und her bewegt werden können, ohne dabei die Verknüpfung zwischen Shows und ihren Media-Tracks zu verlieren. Das ist wichtig, wenn ein Profil auf der SD-Karte eines Raspberry Pi eingerichtet wurde, um es anschließend auf einem USB-Stick im Echtbetrieb einzusetzen. Das gilt genau so für Profile, die auf einem Windowsrechner eingerichtet wurden.

Um diese Portabilität zu erreichen, sollten die Tracks im /pp_home Verzeichnis gespeichert werden, Unterverzeichnisse sind erlaubt. Wenn Tracks unter /pp_home gespeichert werden, bevor das Profil erstellt ist, wird der Profil-Editor automatisch die richtige Verknüpfung erstellen.

- Wenn die Datei unter dem Homeverzeichnis (./pp_home) gespeichert wird, wie das im Menü unter Options > Edit voreingestellt ist, wird die Dateiverknüpfung bei relativem Pfad etwa so aussehen: '+/track_to_play.mp4'.
- Andernfalls sieht die Verknüpfung bei absolutem Pfad folgendermaßen aus: '/home/pi/mymedia/track_to_play.mp4'.

Absolute Pfade machen Sinn, wenn damit beispielsweise Internetadressen wie http://www.mysite.com/track_to_play.mp4. angegeben werden.

4.2.2 Profile auf SD-Karte

Die SD-Karte für Entwicklung und Abspielen zu nutzen ist sehr sinnvoll, wenn Anwendungen auf dem Raspberry Pi erstellt werden

- Erstellen Sie ein Verzeichnis /pp_home im Heimatverzeichnis des Pi-Benutzers und erstellen darin ein Verzeichnis pp_profiles. Sie können auch ein Verzeichnis in /home/pi/pp_home einrichten, etwa unter dem Namen media, um darin Mediendateien abzulegen.
- In den Editor-Einstellungen tragen Sie für das Pi Presents-Daten-Heimatverzeichnis /home/pi/pp_home und für das Startverzeichnis von Medien /home/pi/pp_home/media ein.
- Kopieren Sie die Mediendateien in das /media Verzeichnis.
- Mit dem Editor erstellen Sie ein neues Profil, nennen es myprofile und bearbeiten es.
- Um das Profil zu starten, tippen Sie:
  `python pipresents.py -p myprofile`
  in einem Terminalfenster, das Sie zuvor im pipresents Verzeichnis geöffnet haben.

Wenn Pi Presents mit sudo, durch eine Desktopverknüpfung oder von einer Autostart-Datei gestartet wird, ist es am besten, den vollständigen Pfad von pipresents und auch den vollständigen Pfad des Daten-Heimatverzeichnisses wie folgt anzugeben:

```
sudo python /home/pi/pipresents/pipresents.py -o /home/pi -p myprofile
```

4.2.3 Editor auf einem Windows PC

Der Gebrauch des Editors auf einem Windos PC setzt ein installiertes Python System voraus. Ich

benutze das freie Active Estate Python http://www.activestate.com/activepython
Hier gibt es eine gute Installationsanleitung:
http://www.richarddooling.com/index.php/2006/03/14/python-on-xp-7-minutes-to-hello-world/

Um den Editor auf einem Windows-PC zu installieren, laden Sie die Zip-Datei von https://github.com/KenT2/pipresents-next
(nur auf den Zip-Button klicken). Entpacken Sie den Inhalt des Ordners 'pipresents-master' in einen Ordner namens pipresents.

Die Pi-Presents Beispielprofile pipresents-next-examples entpacken Sie nach dem Herunterladen entweder in Ihren Benutzerordner (C:\users\ken auf meinem Windows 7 Rechner) oder erstellen Sie hier einen 'pp_home'-Ordner mit einem Unterordner 'pp_profiles'.

Öffnen Sie 'pp_editor.py' im Idle-Editor und starten Sie den Editor. Alternativ können Sie ihn mit einem Doppelklick auf 'pp_editor.py' starten. Wenn sie eine Desktop-Verknüpfung für den Start von 'pp_editor.py' eingerichtet haben, müssen Sie die Dateiendung '.py' dem Programm 'python.exe' zuordnen.

Die Bedienung des Editors funktioniert in Windows genauso, wie auf dem Raspberry Pi. Natürlich müssen alle Verknüpfungen der Tracks mit den zugehörigen Mediendateien auf 'pp_home' verweisen.

Auf dem PC erstellte Profile können auf den Pi übertragen werden, mit einem USB-Stick oder alternativ über ein Netzwerk mit einem PC-FTP-Programm wie Filezilla oder WinSCP.

4.2.4 Profile auf USB-Stick

Um Profile von einem USB-Stick zu starten kopieren Sie das 'pp_home'-Verzeichnis von seinem Ort auf einem PC oder dem Pi auf die oberste Ebene eines USB-Sticks.

Stecken Sie den USB-Stick in den Pi und starten Sie Pi Presents mit der -o Option für /media/STICKNAME. STICKNAME ist der Laufwerksname. Windows scheint Kleinbuchstaben eines Laufwerknamens in Großbuchstaben zu verwandeln.

Der Editor könnte auch auf einem Linux-Rechner laufen. Wenn Sie solch ein „Monstrum“ benutzen, brauchen Sie sicherlich keine Hilfe beim Transfer.

5 Die Komponenten in Pi Presents

5.1 Einführung

Shows und Tracks

Das Pi Presents Werkzeug besteht aus zwei Grundelementen – Shows und Tracks

Eine Show zeigt Tracks; die zu spielenden Tracks sind in einer Medialiste enthalten, die der Show zugeordnet ist. Stellen Sie sich die Medialiste als eine erweiterte Playlist vor.

Das Werkzeug bietet aktuell fünf Showtypen, sie eignen sich jeweils besonders gut für

unterschiedliche Zwecke.

- Mediashow – spielt eine Sequenz von Tracks, in der Regel automatisch, allerdings kann das Abspielen auch manuell kontrolliert werden.
- Hyperlinkshow – benutzt Touchscreen-Funktionalität, wie sie in vielen Museen vorkommt.
- Radiobuttonshow – Eine einfache Kioskshow mit einem Navigationsbildschirm – Druck auf einen von mehreren Buttons spielt einen Track.
- Menu – Ähnlich der Radiobuttonshow, aber die Trackauswahl erfolgt mit Durchsuchen eines Menüs mit Pfeiltasten oder einzelnen Buttons
- Liveshow – Wie eine Mediashow, jedoch mit dynamisch eingespieltem Inhalt von außen.

Das Tool Pi Presents hat Player für vier Tracktypen, wobei jeder Player einen anderen Medientyp abspielt. Alle Player erlauben bei Bedarf das Abspielen von Hauptmedien in einem Fenster mit einfachem Farbhintergrund, Bildern und Texten; sie gestatten die Steuerung von Animationen und Start und Stopp bei anderen, konkurrierenden Shows.

- Video – spielt Videos mit dem OMXPplayer.
- Audio – spielt Audiotracks mit dem MPlayer.
- Bilder – zeigt Bilder in vielen Formaten (.jpg etc.).
- Textmeldungen – eine schnelle Möglichkeit, Texte anzuzeigen.

Subshows und Childshows

In Pi Presents kann eine Show ein Track einer Elternshow sein; sie heißt Subshow. Subshows dienen unterschiedlichen Zwecken, wie:

- Aufteilung einer längeren Mediashow in Einzelsegmente. Jedes Segement ist eine Show. Eine Elternshow führt die Segmentshows in ihrer Medialiste auf.
- Ein Menü oder eine Radiobuttonshow können Mediashows als Intro oder als Einzeltracks haben.
- Menüs mit mehreren Ebenen.

Subshows können sich beliebig nach unten verzweigen, eine Grenze bildet dabei möglicherweise nur die Verwirrung des Benutzers, wenn er mit zu vielen Ebenen arbeiten muss. Subshows sind Tracks und tauchen als solche auch in der Medialiste auf.

Childshows sind Subshows, die in besonderer Weise von Mediashows und Liveshows benutzt werden. Wenn Childshows gestartet sind (Has Child) kann eine Childshow so eingerichtet werden, dass sie während des Abspielens eines Tracks in der Elternshow gestartet werden kann. Nach Ablauf der Childshow kehrt die Elternshow zu ihrem nächsten Track zurück, um ihn abzuspielen.

Konkurrierende Shows

Pi Presents kann zwei oder mehr Shows gleichzeitig abspielen. Die konkurrierenden Shows laufen parallel. Das macht Vieles möglich, wie

- einen Audio-Hintergrund für eine Diashow. Gebrauch von zwei Mediashows, eine mit Diashow und die andere mit den Audiotracks.

Bei Letzterem muss die Steuerung abgestellt sein, wenn es eine Nutzer-Interaktion bei Ersterem gibt.
- Shows können zu bestimmten Tageszeiten laufen.
Es laufen zwei oder mehrere Liveshows oder Mediashows, jede auf eine Tageszeit eingestellt. Liveshow und Mediashow können auch zu einer bestimmtern Tageszeit oder nach einer bestimmten Laufzeit gestoppt werden.
- Wirklich sparsam wird es, wenn zwei komplett selbständige Tasks mit einem Pi laufen, - vielleicht eine Diashow im Foyer und eine Erzählstimme in einer Ausstellung, gesteuert mit einer Infrarotfernbedienung.

Jede der konkurrierenden Shows kann Subshows enthalten.

Pi Presents kann so eingerichtet werden, dass jede konkurrierende Show/Subshow ihren eigenen Steuerungssatz besitzt.

5.2 Showtypen

Shows besitzen Felder, in denen ihr Abspielmodus und das Aussehen festgelegt sind. Es gibt einige Felder, die Standardwerte für die Tracks in der Show, wie z.B. Dauer, Überblendung, OMXPlayer und Audio benutzen, wenn ihre entsprechenden eigenen Felder leer sind.

Alle Showes haben folgende Felder

- Title – Text, der im Editor unf den Zeilen einer Menu Show erscheint.
- Show Reference – Ein Name, der anderen Shows als Verweis auf diese Show dient. Erlaubt ist ein alphanumerischer String ohne Leerzeichen.
- Medialist – ist der Name einer Datei, die im Medialistenfenster erscheint. Sie muss die Endung .json haben. Jede Show muss eine Medialiste haben, in der die Tracks dieser Show benannt sind. Die selbe Medialiste kann von mehreren Shows benutzt werden.

5.2.1 Mediashow

Stellen Sie sich eine Mediashow als Diashow vor, die verschiedene Tracktypen abspielen kann – Videos, Audiotracks, Bilder und sogar gesteuerte Animationen.

Mediashows haben Felder, in denen ihre Steuerung festgelegt wird, wie z.B. Startbefehl, Verlauf, Wiederholung und Wiederholungsintervall.

Eine Show kann einer Mediashow so zugeordnet sein, dass die Mediashow von jedem Track der Show aus erreichbar ist.. Das sind 'Child Shows. Das 'Has Child'-Feld teilt Pi Presents mit, dass eine Show dieser Show zugeordnet ist'; die Child Show ist in der Medialiste der Show aufgeführt. Mit einer Child Show ist ein Hinweistext verbunden, der nur angezeigt wird, wenn 'Has Child' ist 'Yes'.

Mediashows haben mehrere Optionen im Repeat-Feld, die ihren Ablauf steuern

- interval – Die Mediashow wiederholt sich, bis sie gestoppt wird.
- Oneshot – Die Mediashow wartet auf einen Startbefehl, läuft dann einmal und wartet auf einen neuen Startbefehl. Sie wird durch andere Befehle gestoppt.

- Single-run – Die Mediashow läuft einmal und stoppt dann von selbst. Es ist kein ausdrücklicher Stopp-Befehl erforderlich, obwohl auch das möglich ist
- Show Control - Stopp-Befehle für eine Show, die bereits gestoppt ist, werden ignoriert. Wird hauptsächlich in Show Control gebraucht.

Unmittelbar nach dem Start erwarten alle Mediashows Anweisungen für ihre Fortsetzung.

Mediashows mit dem Startbefehl = start laufen sofort. Mediashows mit dem Startbefehl = input fahren fort, wenn die Steuerungseingabe erfolgt.

Weitere Informationen über die Kontrolle von Mediashows in Kapitel 6.2.1

Field	Examples	Values
		Wichtige Informationen
Type	mediashow	Kann nicht verändert werden
Title	My First Show	Text beschreibt die Show. Wird im Editor und in Menüs angezeigt.
Show Reference	show1	Eintrag, wie die Show mit anderen Shows verknüpft ist. Text ohne Leerzeichen.
Medialist	mymedia.json	Dateiname der Medialiste, die die Tracks für die Mediashow enthält.
Show Canvas		Nicht aktiv
		Wie die Show laufen soll
Disable Controls	no	yes/no Wenn 'yes' dann sind interne Operationen (Start, Pause, Auf, Ab, Stopp) und Laufzeitkontrollen deaktiviert. Dies ist eine schnelle Möglichkeit, alle Steuerungen für konkurrierende Shows zu sperren. Für eine feinere Steuerung stehen die 'controls'-Felder zur Verfügung. (siehe Kapitel 6.4)
Trigger For Start	start	Wie die Mediashow nach dem Start fortgesetzt wird: • start – läuft ohne Halt • input/input-quiet – wartet auf eine Eingabe. 'input-quiet' schaltet Bildschirmnachrichten ein. • time/time-quiet – wartet auf eine Tageszeit. 'time-quiet' schaltet den nächsten Showtext ein.

Field	Examples	Values
Trigger Input	PIR	Wenn der Startbefehl 'input' oder 'input-quiet' ist: Ein symbolischer Name wird mit einer Eingabe verknüpft, wie in Kapitel 10.2 beschrieben. Wenn der Startbefehl 'time' ist: Eine Liste von Tageszeiten h:m:s im 24-Stundenformat, getrennt durch Leerzeichen (siehe Kapitel 6.2.1).
Progress	auto	Wie die Show zwischen Tracks fortgesetzt wird: • manual - Pi presents wartet auf Auf/Ab interne Befehle zwischen den Tracks. • auto – Tracks laufen kontinuierlich, mit Auf/Ab internen Befehlen abbrechen.
Trigger For Next	continue	Nächsten Track steuern, wenn Progress 'manual' ist: • input – wartet auf Eingabe • continue – wartet nicht
Next Input		Wenn Trigger For Next ist 'input': Ein symbolischer Name ist mit einer Eingabe verknüpft, wie in Kapitel 10.2. beschrieben.
Sequence		Reihenfolge von Tracks: • ordered – werden in der Reihenfolge der Medialiste abgespielt. • shuffle – Abspiel in zufälliger Reihenfolge. Unter 'shuffle' enden Tracks nicht selbständig. Sie können mit 'Trigger For End' beendet werden.

Field	Examples	Values
Repeat	interval	Wie eine Mediashow wiederholt wird: • oneshot – Tracks werden einmal gespielt. Danach wartet die Show auf einen Startbefehl. • interval – Abspielen in Intervallen • single-run – Mediashow läuft einmal und stoppt. Wenn das Wiederholungsintervall erreicht ist, werden laufende Tracks nicht unterbrochen. Deshalb müssen alle Tracks in einer sich wiederholenden Show 'non-zero-duration' haben oder bis zum Ende laufen. Um einen leeren Bildschirm nach Ablauf einer Mediashow zu bekommen, kann ein leerer Message-Track von 1 Sekunde Dauer eingefügt werden.
Repeat Interval	10	Die Zeit zwischen dem Start einer Mediashow und dem Start der Nächsten in Sekunden: • Wenn der Wert kleiner ist als die Länge der Show, wird die Show wiederholt abgespielt. • Der Wert 0 bewirkt immer ein wiederholtes Abspielen.
Trigger For End		Wie das Ende einer Show gesteuert werden kann: • none – Ein Ende-Befehl ist nicht möglich. • Time – Beendet die Show zu einer bestimmten Tageszeit. • duration – Beendet die Show nach einer bestimmten Dauer.
End Times	5:00	Ende-Befehl = time: • Eine Liste von Tageszeiten im 24-Stundenformat, durch Leerzeichen getrennt. (siehe Kapitel 6.2.1). Ende-Befehl = duration • ein Zeitraum im h:m:s - Format.

Field	Examples	Values
		Tracks einer Mediashow können einen Einstiegspunkt für eine Show oder einen Track zur Verfügung stellen. Mit dem Startvorgang wird die Show gestartet. Die Mediashow wird fortgesetzt, wenn die Subshow beendet ist. **Wenn Has Child = yes wird der Hinweis angezeigt.**
Has Child	yes	yes/no. Mit 'yes' ist die 'Child Show'-Funktion aktiviert. ChildShow/Track ist in der Medialiste mit der Trackreferenz pp-child-show enthalten.
Hint Text	Press Play to..	Einzelne Textzeile
Hint Font	Helvetica 30 bold	Eine Schrift mit der Schriftauswahl wählen oder Schrift direkt eintragen: Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm
Hint Colour	white	Schriftfarbe im Auswahlfenster wählen, erzeugt eine sechsstellige Hexadezimalzahl, oder Farbnamen direkt eintragen. Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm
Hint x	500	Textposition von links
Hint y	900	Textposition von oben
		Show Text erscheint als Overlay in allen Tracks - Message Tracks dagegen nicht.
Show Text	Picture of Taj Mahal	Angezeigter Text, wenn Feld nicht leer.
Show Text Font	Helvetica 30 bold	Eine Schrift mit der Schriftauswahl wählen oder Schrift direkt eintragen: Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm
Show Text Colour	white	Schriftfarbe im Auswahlfenster wählen, erzeugt eine sechsstellige Hexadezimalzahl, oder Farbnamen direkt eintragen. Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm
Show Text x Position	100	Entfernung des Textbeginns vom linken Bildschirmrand (Pixel)
Show Text y Position	100	Entfernung des Textes vom oberen Bildschirmrand (pixel).

Field	Examples	Values
Controls		Wenn diese Show an der Spitze der Show/Subshow-Liste steht, werden Verknüpfungen mit internen und Laufzeit-Funktionen in der controls.cfg überschrieben. Siehe Kapitel 10.2.4
		Tracks in einer Show benötigen einige Einstellungen, wenn sie nicht schon in den Trackeinstellungen enthalten sind.
Background Image	+/media/image.jpg	Wenn das Feld nicht leer ist steht hier der Dateiname des Bildes, das als Hintergrundbild für Tracks in der Show dient.
Background Colour		Schriftfarbe im Auswahlfenster wählen, erzeugt eine sechsstellige Hexadezimalzahl, oder Farbnamen direkt eintragen. Siehe http://effbot.org/tkinterbook/tkinter-widget-styling.htm Die Hintergrundfarbe ist beim Start von Pi Presents auf schwarz gesetzt. Der Wert in diesem Feld wird von einem Track benutzt, wenn das entsprechende Feld im Track leer ist. Wenn das resultierende Feld Background Colour leer ist, bleibt die Hintergrundfarbe unverändert
Transition	cut	Cut - Typ der Überblendungen zwischen den Tracks Aktuell nicht genutzt.
Duration	10	Sekunden - Spieldauer eines Tracks, der kein reguläres Ende hat. Wert 0 bewirkt wiederholtes Abspielen.
Image Window	original 10 100 fit fit 1010 40 40 fit ANTIALIAS	Ein Sichtfenster für Bilder einer Show. Siehe Kapitel 5.4.4
Video Audio	hdmi	hdmi/local/<blank>. Tonausgabe-Kanal für Video-Tracks einer Show über den OMXPlayer. Wenn das Feld leer bleibt, erfolgt die Tonausgabe automatisch über vorhandenen HDMI-Monitor.
Video Audio Volume	0	Lautstärke des Videotons (-60 -> 0 dB).
Video Window	original warp	Für VideoTracks der Show ein Sichtfenster, in dem das Video läuft. • original – OMXPlayer

Field	Examples	Values
	warp 10 100 200 700	Standardverhalten. • warp – skaliert auf Bildschirmgröße ohne das Bildseitenverhältnis zu verändern. • warp, gefolgt von zwei x,y-Paaren (oben links, unten rechts) für das Sichtfenster. Das Video wird auf diese Fenstergröße skaliert ohne das Bildseitenverhältnis zu bewahren.
Video Player Options		Weitere Optionen für den omxplayer (Vorsicht ist angebracht, sonst: „Have a nice day!“)
Audio Player Audio	local	hdmi/local. Tonausgabekanal für die Audiotracks der Show, die der MPlayer abspielt.
Audio Player Speaker	stereo	left/right/stereo. Lautsprecher für alle Audiotracks der Show, die der MPlayer abspielt.
Audio Player Volume	0	(-60 -> 0dB) - Lautstärke der Audiotracks der Show, die der MPlayer abspielt.
Audio Player Options		Weitere Optionen für den MPlayer (Vorsicht ist angebracht um Neustart zu vermeiden!)
Web Window	warp warp 10 100 200 700	Für jeden Web-Track der Show ein Sichtfenster, in dem die Webpage läuft. • warp – skaliert die Webseite auf Bildschirmgröße ohne das Bildseitenverhältnis zu verändern. • warp, gefolgt von zwei x,y-Paaren (oben links, unten rechts) für das Sichtfenster. Die Webseite wird auf diese Fenstergröße skaliert ohne das Bildseitenverhältnis zu bewahren.

5.2.2 Menü

Eine Menü-Show benutzt die Titel und Miniaturfotos von Tracks und Shows, um automatisch ein Menü auf dem Bildschirm zu generieren. Mit den internen 'Auf'/'Ab'-Funktionen ist das Scrollen im Menü möglich. Mit der internen 'Play'-Funktion können Tracks oder Shows abgespielt werden.

Mit den Feldern in der Feld-Tabelle des Editors wird das Aussehen des Menüs gestaltet:

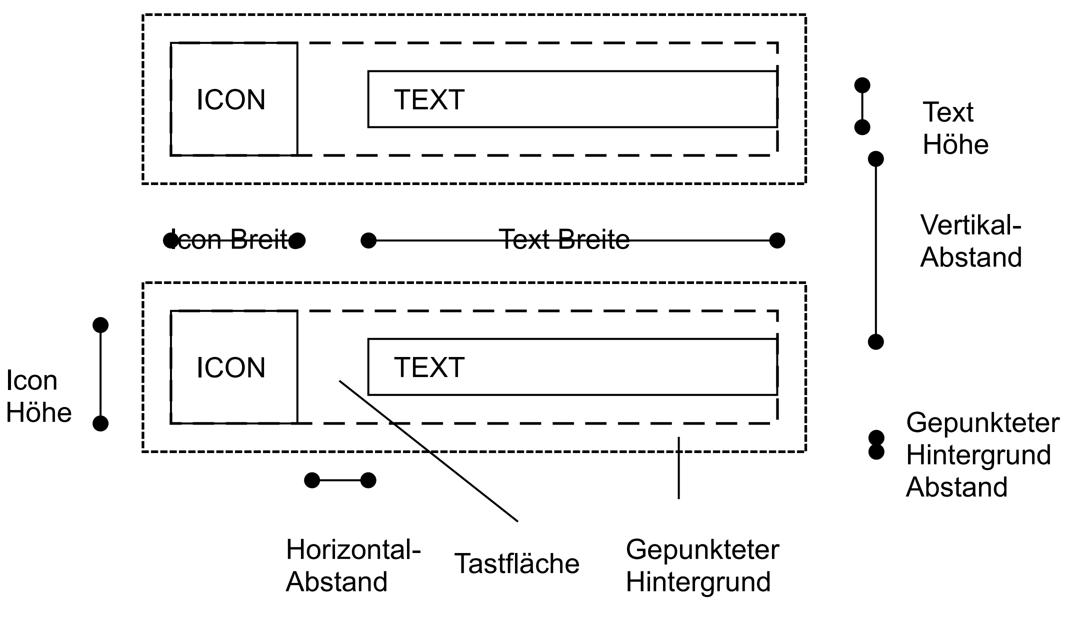
- Bildschirm, Miniaturbilder, Schaltknöpfe oder Titel in vielen Kombinationen
- Vertikales und horizontales Durchstreifen des Menüs

- Einrichten von Menüs mit mehreren Spalten und Zeilen
- Ändern der Abstände, Farbe und Schriften der Menüpunkte

Zusätzlich zu den Menüeinträgen können zwei zusätzliche Textfelder, Hinweistext und Menütext, beliebig auf dem Bildschirm platziert werden.

Wenn ein Menü einige Zeit nicht gebraucht wird, kann es sinnvoll sein, zur Elternshow zurückzukehren. Das funktioniert, wenn das Timeout-Feld nicht '0' ist.

Ein Menü kann ein Hintergrundbild besitzen. Das wird im Background-Feld festgelegt. Der Speicherort der Bilddatei ist in der Medialiste mit der Show verknüpft.



Menü-Aufteilung

Field	Examples	Values
		Wichtige Informationen
Type	menu	Kann nicht verändert werden
Title		Text beschreibt die Show und wird im Editor angezeigt.
Show Reference	mymenu	Bezeichnung, unter der andere Shows sich auf diese Show beziehen. Text ohne Leerzeichen.
Show Canvas		Wird nicht verwendet
Medialist	themenu.json	Dateiname der Medialiste, die die Tracks für das Menü enthält.
		Das Aussehen des Menüs

Field	Examples	Values
Entry Font	Helvetica 30 bold	Eine Schrift mit der Schriftauswahl wählen oder Schrift direkt eintragen: Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm
Entry Colour	white	Schriftfarbe im Auswahlfenster wählen, erzeugt eine sechsstellige Hexadezimalzahl, oder Farbnamen direkt eintragen. Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm
Selected Entry Colour	red	Farbe, wenn der Eintrag gewählt ist.
Timeout	60	Sekunden - Wenn das Menü nicht benutzt wird, kehrt Pi Presents automatisch zum vorherigen Bildschirm zurück. '0' für keine Auszeit.
Menu Background Colour		Values siehe mediashow - Background Colour.
Has Background Image	yes	yes/no. Bei 'yes' wird ein Bild hinter dem Menü angezeigt. Das Bild ist in der Medialiste mit der Track-Referenz pp-menu-background aufgeführt.
		Menü-Aussehen und Einteilung
Menu Window		fullscreen - ein oder zwei Paare x,y-Koordinatenfelder durch Leerraum getrennt. fullscreen – das Menü erscheint als Ganzbildschirm. ein x,y,-Koordinatenpaar bezeichnet obere linke Ecke des Menüs. zwei x,y,-Koordinatenpaare – das zweite Paar bezeichnet die rechte untere Ecke einer Anzeigebox für das Menü.
Direction	vertical	vertical, horizontal – Anordnung der Menüeinträge und der Suchrichtung.
Rows	5	Wenn Direction = vertical – Zahl der Zeilen des Menüs.
Columns	1	Wenn Direction = horizontal – Zahl der Spalten des Menüs.

Field	Examples	Values
Icon Mode		Thumbnail, bullets, none thumbnail – wenn das Track ein Miniaturbild hat, wird es dargestellt, oder wenn es ein Image Track ist wird es auf die Icongröße reduziert, oder ein Miniaturbild wird gezeigt, das hängt vom Tracktyp ab. bullet- das im Bullett-Feld enthaltene Icon wird angezeigt. none – Das Menü zeigt nur Text an.
Text Mode		None, right, below right – Der Text wird rechts neben dem Icon angezeigt. below – Der Text wird unter dem Icon angezeigt.
Bullet	+/media/bullet.jpg	Der Dateiname des Bullets; absolut oder relativ.
Icon Width	100	Breite des Icons, Bilder werden auf die eingestellte Größe verändert.
Icon Height	100	Höhe des Icons, Bilder werden auf die eingestellte Größe verändert.
Horizontal Padding	10	Horizontaler Abstand zwischen Icon und Text, wenn der Textmodus 'right' ist.
Vertical Padding	10	Vertikaler Abstand zwischen Icon und Text, wenn der Textmodus 'below' ist.
Text Width	200	Textbreite, der Text wird in diese Breite eingepasst.
Text Height	50	Texthöhe, Berechnet die Position der Menüeinträge. Der Text wird nicht in dieser Höhe eingepasst. Dafür muss die Schriftgrösse angepasst werden.
Horizontal Separation	20	Der Abstand zwischen dem unteren Rand eines Menüeintrages und dem oberen Rand des nächsten Menüeintrages.
Vertical Separation	20	Der Abstand zwischen dem rechten Rand eines Menüeintrages und dem linken Rand des nächsten Menüeintrages.
Stipple Background		yes/no. Zeigt einen gepunkteten rechteckigen Rahmen hinter dem Menüeintrag.
Stipple Background Padding		Der Abstand des gepunkteten Hintergrundrahmens vom Menüeintrag.

Field	Examples	Values
Guidelines		Never, auto, always Wenn das Menüfenster zwei x,y,-Paare besitzt, werden farbige Rechtecke eingeblendet, die bei der Einrichtung eines Menüs helfen. • Weiße Rechtecke umfassen die Menüeinträge. • Blaues Rechteck umfasst die kalkulierte Größe des Menüs. • Rotes Rechteck umfasst die vom Menüfenster beschriebene Rahmenbox. Auto zeigt die Rechtecke nur, wenn das Menü größer als die Rahmenbox ist.
		Textfelder des Menüs
Menu Text	This menu	Mehrzeilentext
Menu Text Font	Helvetica 30 bold	Eine Schrift mit der Schriftauswahl wählen oder Schrift direkt eintragen: Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm
Menu Text Colour	white	Schriftfarbe im Auswahlfenster wählen, erzeugt eine sechsstellige Hexadezimalzahl, oder Farbnamen direkt eintragen. Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm
Menu Text x	100	Entfernung des Textbeginns vom linken Bildschirmrand (Pixel)
Menu Text y	800	Entfernung des Textes vom oberen Bildschirmrand (pixel).)
Hint Text	To Play..	Menüs besitzen keine 'Children'; 'hint' ist eine allgemeine Möglichkeit, um Abspielanweisungen zu erteilen.
Hint Font	Helvetica 30 bold	Eine Schrift mit der Schriftauswahl wählen oder Schrift direkt eintragen: Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm
Hint Colour	white	Schriftfarbe im Auswahlfenster wählen, erzeugt eine sechsstellige Hexadezimalzahl, oder Farbnamen direkt eintragen. Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm

Field	Examples	Values
Hint x	100	Entfernung des Textbeginns vom linken Bildschirmrand (Pixel).
Hint y	800	Entfernung des Textes vom oberen Bildschirmrand (pixel).
		Show Text erscheint als Overlay auf allen Tracks, im Gegensatz zu Message Tracks, die Vom Menü aus geöffnet werden.
Show Text	Picture of Taj Mahal	Wenn nicht leer, wird der Text angezeigt.
Show Text Font	Helvetica 30 bold	Eine Schrift mit der Schriftauswahl wählen oder Schrift direkt eintragen: Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm
Show Text Colour	white	Schriftfarbe im Auswahlfenster wählen, erzeugt eine sechsstellige Hexadezimalzahl, oder Farbnamen direkt eintragen. Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm
Show Text x Position	100	Entfernung des Textbeginns vom linken Bildschirmrand (Pixel)
Show Text y Position	100	Entfernung des Textes vom oberen Bildschirmrand (pixel).
		Tracks in einer Show benötigen einige Einstellungen, wenn sie nicht schon in den Trackeinstellungen enthalten sind.
Background Image	+/media/image.jpg	Values siehe Mediashow
Background Colour		Values siehe Mediashow
Transition	cut	Values siehe Mediashow
Duration	10	Values siehe Mediashow
Image Window		Values siehe Mediashow
Video Player Audio	hdmi	Values siehe Mediashow
Video Player Volume	0	Values siehe Mediashow
Video Window		Values siehe Mediashow
Video Player Options	-t 1	Values siehe Mediashow
Audio Player Audio	local	Values siehe Mediashow
Audio Player Speaker	stereo	Values siehe Mediashow
Audio Player Volume	0	Values siehe Mediashow
Audio Player Options		Values siehe Mediashow
Web Window		Values siehe Mediashow
Browser Options		Values siehe Mediashow

Field	Examples	Values
		Steuerungen
Disable Controls	no	yes/no Wenn 'yes' dann sind interne Operationen (Start, Pause, Auf, Ab, Stopp) und Laufzeitkontrollen deaktiviert. Dies ist eine schnelle Möglichkeit, alle Steuerungen für konkurrierende Shows zu sperren. Für eine feinere Steuerung stehen die 'controls'-Felder zur Verfügung. (siehe Kapitel 6.4).

5.2.3 Liveshow

Eine Liveshow ist eine Reihenfolge von einem oder mehreren Bildern, Videos so wie Webseiten und Audiotracks. Die Inhalte werden dynamisch eingespeist.

Liveshows zeigen Tracks nacheinander wiederholt in alphabetischer Reihenfolge ihrer Dateinamen.

Mit der Liveshow können eine Show oder ein Track so verbunden sein, dass die Show von jedem Track der Show erreichbar ist. Dies sind Child Shows. Das Feld 'Has Child' signalisiert Pi Presents dass eine Child Show mit der Show verbunden ist; Die Show selbst ist in der Medialiste mit der Liveshow verknüpft. Mit dem Einsatz einer Child Show ist 'Hint Text' verfügbar, der Hinweistext wird angezeigt, wenn 'Has Child' = 'yes'.

Die mit der Liveshow verbundene Medialiste beschreibt die Child Show, unbenannte Tracks werden ignoriert.

Liveshows laufen, bis sie gestoppt werden. Unmittelbar nach dem Start entscheiden Sie über folgende Fortsetzungsbedingungen:

- Liveshows mit 'Trigger For Start' = 'start' starten sofort. Liveshows mit 'Trigger For Start' = 'time' fahren fort, wenn die Startbedingung erfüllt ist. Die Liveshow-Steuerung ist ausführlicher erklärt in Kapitel 6.2.1

Field	Examples	Values
		Wichtige Informationen
Type	liveshow	Kann nicht verändert werden
Title	My Live Show	Text beschreibt die Show. Wird im Editor und in Menüs angezeigt.
Show Reference	myliveshow	Eintrag, wie die Show mit anderen Shows verknüpft ist. Text ohne Leerzeichen.
Show Canvas		Nicht aktiv

Field	Examples	Values
Disable Controls	no	yes/no Wenn 'yes' dann sind interne Operationen (Start, Pause, Auf, Ab, Stopp) und Laufzeitkontrollen deaktiviert. Dies ist eine schnelle Möglichkeit, alle Steuerungen für konkurrierende Shows zu sperren. Für eine feinere Steuerung stehen die 'controls'-Felder zur Verfügung. (siehe Kapitel 6.4).
Trigger For Start	time	start/time/time-quiet Wie die Liveshow sich nach dem Start fortsetzt:: • start – weiter ohne warten • time/time-quiet – wartete auf Tageszeit. time-quiet unterdrückt die nächste show message.
Start Times	9:30 2:30	Eine Liste mit Tageszeiten h:m:s im 24 Stundenformat, getrennt durch Leerzeichen. (Siehe Kapitel 6.2.1).
Trigger For End	none	none/time Steuerung des Show-Endes • none – wird immer wiederholt • time – beendet zu einer Tageszeit
End Times	12:00 17:30 5	Wenn Trigger For End = time: • Eine Liste mit Tageszeiten h:m:s im 24-Stundenformat, getrennt durch Leerzeichen (siehe: Kapitel 6.2.1). Wenn Trigger For End = duration • ein Zeitraum im h:m:s Format.
Medialist	mymedia.json	Dateiname der Medialiste, die den Child Show Track für die Liveshow enthält. Muss vorhanden sein, auch wenn es keine Child Show gibt.
		Liveshows können einen Einstiegspunkt für eine Show anbieten. Wenn die interne Operation 'play' ausgeführt wird, startet die Show. Die Liveshow wird fortgesetzt, wenn das Menü beendet ist.
Has Child	yes	yes/no. Wenn 'yes' ist die 'child'-Eigenschaft aktiviert. Eine Child Show ist in der Medialiste mit der Track-Referenz pp-child-show eingerichtet.
Live Tracks Directory		Verzeichnis, das Tracks für diese Show enthält. (Für künftige Erweiterung).

Field	Examples	Values
		Ein Hinweis ist eine Zeile Text, die nahe des unteren Bildschirmrandes angezeigt wird, wenn 'Has Child' = 'yes'. Hinweis weist auf die Existenz eines Menüs hin und erklärt, wie man das Menü startet.
Hint Text	Press Play to..	
Hint Font	Helvetica 30 bold	Eine Schrift mit der Schriftauswahl wählen oder Schrift direkt eintragen: Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm
Hint Colour	white	Schriftfarbe im Auswahlfenster wählen, erzeugt eine sechsstellige Hexadezimalzahl, oder Farbnamen direkt eintragen. Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm
Hint x	200	Textposition von links
Hint y	900	Textposition von oben
		Show Text liegt auf allen Bildern der Show.
Show Text	Picture of Taj Mahal	Wenn nicht leer, erscheint der Text bei allen Bildern, Videos und Audiotracks der Show
Show Text Font	Helvetica 30 bold	Eine Schrift mit der Schriftauswahl wählen oder Schrift direkt eintragen: Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm
Show Text Colour	white	Schriftfarbe im Auswahlfenster wählen, erzeugt eine sechsstellige Hexadezimalzahl, oder Farbnamen direkt eintragen. Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm
Show Text x Position	100	Textabstand vom linken Bildschirmrand (Pixel)
Show Text y Position	100	Textabstand vom oberen Bildschirmrand (Pixel).
Controls		Values siehe Mediashow
		Tracks in der Show benötigen Einstellungen.
Background Image	+/media/image.jpg	Values siehe Mediashow
Background Colour		Values siehe Mediashow
Transition	cut	Values siehe Mediashow
Duration	10	Values siehe Mediashow
Image Window		Values siehe Mediashow
Video Player Audio	hdmi	Values siehe Mediashow

Field	Examples	Values
Video Player Volume	0	Values siehe Mediashow
Video Window		Values siehe Mediashow
Video Player Options		Values siehe Mediashow
Audio Player Audio	local	Values siehe Mediashow
Audio Player Speaker	stereo	Values siehe Mediashow
Audio Player Volume	0	Values siehe Mediashow
Audio Player Options		Values siehe Mediashow

5.2.4 Radiobuttonshow

Eine Radiobuttonshow verfügt über Präsentationsmöglichkeiten, wie sie in vielen Kiosks üblich sind:

> 'Startet mit einem Einstiegsbildschirm, mit einem Text, der den Benutzer zum Druck auf einen Button auffordert, der einen Track startet. Druck auf einen Button spielt einen Track, der diesem Button zugeordnet ist. Beim Abspielen des Tracks startet ein Druck auf einen anderen Button dessen zugeordneten Track. Am Ende eines Tracks oder durch Druck auf Stopp erfolgt die Rückkehr zum Einstiegsbildschirm.'

In Pi Presents kann der Einstiegsbildschirm ein Bild, eine Nachricht, ein Video, ein Audiotrack oder eine Show sein. Weitere Tracks werden nach Maßgabe der Play Befehle im Links Feld der Radiobuttonshow gespielt, z.B.:

```
but1 play myimagetrack
but2 play myvideotrack
```

Jeder Befehl hat drei durch Leerzeichen getrennte Felder:

- symbolic name – der symbolische Name der Eingabe-Aktion, die das Abspielen von Shows oder Tracks steuert.
- command – ist immer 'play' in einer Radiobuttonshow. Nicht das Gleiche, wie die interne Play-Operation.
- track – Die Track Referenz (Speicherort) des zu spielenden Tracks.

Die internen Operationen werden in der Radiobuttonshow eigentlich nicht gebraucht. Wenn die Radiobuttonshow jedoch einen Track abspielt oder eine Show startet, könnten die internen Operationen doch erforderlich sein.

Field	Examples	Values
		Wichtige Informationen
Type	radiobuttonshow	Kann nicht verändert werdenb
Title	My Show	Text beschreibt die Show. Wird im Editor und in Menüs angezeigt.
Show Reference	myradioshow	Eintrag, wie die Show mit anderen Shows verknüpft ist. Text ohne Leerzeichen.
Show Canvas		Nicht aktiv

Field	Examples	Values
Medialist	mymedia.json	Dateiname der Medialistendatei, die die Tracks für die Radiobuttonshow enthält. Alle Tracks sollten eine Trackreferenz haben
Links	myname play mytrack	Siehe oben.
First Track	myfirsttrack	Die Track Referenz des Tracks, der den Einstiegsbildschirm der Show gestaltet.
Disable Controls	no	yes/no Wenn 'yes' dann sind interne Operationen (Start, Pause, Auf, Ab, Stopp) und Laufzeitkontrollen deaktiviert. Dies ist eine schnelle Möglichkeit, alle Steuerungen für konkurrierende Shows zu sperren. Für eine feinere Steuerung stehen die 'controls'-Felder zur Verfügung. (siehe Kapitel 6.4)
Timeout	30	Sekunden - Wenn kein Benutzerbefehl anliegt oder der Track nicht vor dem Timeout sowieso endet, erfolgt die Rückkehr zum Einstiegstrack.
		Show Text liegt auf allen Bildern der Show.
Show Text	Picture of Taj Mahal	Wenn nicht leer, erscheint der Text bei allen Bildern, Videos und Audiotracks der Show.
Show Text Font	Helvetica 30 bold	Eine Schrift mit der Schriftauswahl wählen oder Schrift direkt eintragen: Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm
Show Text Colour	white	Schriftfarbe im Auswahlfenster wählen, erzeugt eine sechsstellige Hexadezimalzahl, oder Farbnamen direkt eintragen. Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm
Show Text x Position	100	Textabstand vom linken Bildschirmrand (Pixel).
Show Text y Position	100	Textabstand vom oberen Bildschirmrand (Pixel).
Controls		Values siehe Mediashow
		Tracks in der Show benötigen Einstellungen.
Background Image	+/media/image.jpg	Values siehe Mediashow
Background Colour		
Transition	cut	Values siehe Mediashow
Duration	10	Values siehe Mediashow
Image Window		Values siehe Mediashow
Video Player Audio	hdmi	Values siehe Mediashow
Video Player Volume	0	Values siehe Mediashow
Video Window		Values siehe Mediashow

Field	Examples	Values
Video Player Options		Values siehe Mediashow
Audio Player Audio	local	Values siehe Mediashow
Audio Player Speaker	stereo	Values siehe Mediashow
Audio Player Volume	0	Values siehe Mediashow
Audio Player Options		Values siehe Mediashow
Web Window		Values siehe Mediashow
Browser Options		Values siehe Mediashow

5.2.5 Hyperlinkshow

Eine Hyperlinkshow bietet Show-Eigenschaften an, wie sie auf Touchscreens in Museen benutzt werden:

> 'Start mit einer Einstiegsseite mit etwas einführendem Text, Video oder Bildern so wie einer Auswahl von Buttons, die es dem Benutzer ermöglichen, zu anderen Seiten zu wechseln. Jede Seite kann über eine andere Buttonauswahl verfügen, um andere Seiten aufrufen zu können. Auf allen Seiten, außer der Startseite, erlauben zusätzliche Buttons, auf vorherige Seiten zurückzugehen oder die Startseite wieder aufzurufen.'

Alle Touchscreen-Displays, die ich mir in den vergangenen Monaten angeschaut habe, scheinen in dieser Weise zu funktionieren, es sei, dass es sich um Radiobuttonshows handelt.

Jede Seite in einer Hyperlinkshow ist ein Track. Alle Seiten haben Verknüpfungsfelder, die Befehle für die Bewegung zwischen Tracks und zurück enthalten:

z.B. könnte track story1b enthalten:

```
next-name call story1c
alternative-name call alternative1
back-name return
home-name home
```

xxx-name ist der symbolische Name einer Eingabe. Das heißt in diesem Fall, gehe entweder vorwärts zu story1c oder zu alternative1, zurück zum vorherigen Track (möglicherweise story1b) oder zurück zum Home-Track, was eventuell heißt, zurück über story1b und story1a, ohne diese beiden Tracks anzuzeigen.

Wenn der 'call'-Befehl ausgeführt wird, merkt sich Pi Presents, über welchen Pfad der Befehl gekommen ist (zumeist ein Ordnerstapel) So kann der 'return'-Befehl eine Seite zurück gehen, indem die aktuelle Seite im Pfad gelöscht wird, vergleichbar mit verschachteltem Aufruf und Zurück (return) bei Unterprogrammen in Programmiersprachen.

Es gibt eine besondere Funktion genannt Home Track. Das Kommando home führt längs des gewählten Pfades zurück bis zum Home Track. Es ist nicht mehr nötig, den genauen Pfad zu einem bekannten Startpunkt zu kennen.

Jeder Link hat drei Felder, getrennt von Leerräumen:

- symbolic name – der symbolische Name der Eingabe-Aktion, die den Befehl 'command' steuert
- command - call, return, goto, jump, exit, null, repeat
- track - die Track Referenz des Tracks, der gespielt werden sollen

Command	Argument	Effect
call	Track Reference	spielt den Track (nach Reference) und merkt sich den Pfad
return	<blank>	return 1 ruft den Pfad auf, entfernt daraus den Track und endet bei Home Track.
return	number	return n tracks ruft den Pfad auf, entfernt daraus den Track und endet bei Home Track.
return	<Track Reference>	return - zur angegebenen Track Referenz, entfernt die Tracks aus dem Pfad und geht über Home Track, wenn erforderlich, zurück.
home		return – folgt dem Pfad zum Home Track, vorherige Tracks werden aus dem Pfad entfernt
jump	<Track Reference>	play - Track Reference, löscht den Pfad zurück zum Home Track
goto	<Track Reference>	Play - Track Reference, 'vergisst' den Pfad
repeat	<blank>	Wiederholt den aktuellen Track ohne ein Zeitintervall zurückzusetzen
null		Inhibits – sperrt den Link mit dem gleichen symbolischen Namen, der in der Show eingetragen ist.
exit		end – beendet die Hyperlinkshow

Im Links Feld der Hyperlinkshow können Befehle zusätzlich zum Links Feld der Tracks gespeichert werden. Das erspart Schreibarbeit, weil die Befehle in der Show in jedem Track genutzt werden und die Befehle der Tracks mit ihnen zusammengelegt werden. Das ist manchmal nicht erwünscht, so kann man mit dem 'null' Befehl im Track den Befehl mit dem gleichen symbolischen Namen löschen.

Der 'goto' Befehl „weiß“ nicht, wo er herkam. Er wird für besondere Umstände wie zum Beispiel timeout benötigt. Er ist zudem eine Alternative zu 'call/return', um einen Zurück-Button einzufügen; zurück wird dadurch zu 'goto'. Es wäre nicht gut, im gleichen Teil einer Hyperlinkshow 'call/return' und 'goto' zu vermischen.

Wenn ein Track, wie video oder timed image, von selbst endet, muss Pi Presents einen Befehl erteilen, der angibt, was als Nächstes erfolgen soll. Das wird in Pi Presents durch eine interne Eingabe mit dem symbolischen Namen pp-onend erreicht. Das Links Feld hat damit einen Befehl, der ausgeführt werden kann, z.B.

pp-onend goto mynextpage

oder
pp-onend repeat

Ein zweiter spezieller Track ist der First Track. PiPresents beginnt die Show immer hier.. In den meisten Anwendungen haben Home Track und First Track die gleiche Trackreferenz, sie können dennoch unterschiedlich sein. Das ist so eingerichtet, dass First Track die Show mit einem Bild oder Video startet, um den Benutzer dazu zu verlocken, mit einem Link zum Home Track zu springen, um interaktiv eine Reihe von Tracks zu starten. Einmal verpasst würde home return oder jump den User nicht zurück zum First Track bringen.
Mit timeout funktioniert es jedoch.

Um sicher zu stellen, dass das Bild zum First Track zurück, wenn der Benutzer das Bild verlässt, gibt es timeout. Wenn timeout eingeschaltet ist, geht die Hyperlinkshow zum Timeout Track. Dieser Track hat die gleiche Trackreferenz wie der First Track, aber ein eigenständiger Timeout Track erlaubt zum Beispiel das Beenden, bevor es zum First Track geht.

Tracks in Hyperlinkshows können Shows sein. Sie können jedoch keine eigenen Links besitzen, erben sie aber von der Hyperlinkshow, was nicht besonders hilfreich ist.

Auch wenn die internen Operationen in der Hyperlinkshow nicht benutzt werden, könnten sie dennoch beim Abspielen von Tracks oder Starten von Shows in der Hyperlinkshow erforderlich sein und stehen zur Verfügung..

Beim Entwickeln einer Hyperlinkshow ist es oft nützlich, den Pfad für die Webseiten zu sehen, den Pi Presents sich gemerkt hat. Dafür muss man in der Datei pp_hyperlinkshow.py, ungefähr bei Zeile 88, # sign from self.debug = True durch Löschen des #-Zeichens am Anfang der Zeile auskommentieren.

5.2.5.1 Klickbereiche

Im Pilotprogramm einer Hyperlinkshow, das ich gemeinsam mit einem Museum entwickelt habe, hatten wir Softbuttons oder einen Touchscreen zur Steuerung der Hyperlinkshow vorgesehen. Beide benötigen Klickbereiche.

Pi Present ermöglicht in seiner screen.cfg Datei die Festlegung von Klickbereichen. Das sind vieleckige Bereiche des Bildschirms, die auf Berührung oder einen Mausklick reagieren. Eine Berührung oder ein Klick bewirken eine Eingabeaktion, die mit ihrem symbolischen Namen identifiziert wird.

Klickbereiche können Text, Hintergrundfarbe und Umrandungen haben. Die auf jeder Seite angezeigten Klickbereiche sind durch die symbolischen Namen in der Link-Liste der zusammen gehörenden Show- und Tracklinks bestimmt.

Eine Hyperlinkshow kann auch mit der Tastatur oder mit Buttons gesteuert werden. In unserem Pilotprogramm wollten wir Softbuttons benutzen, eine Technik, die ich auf einem Computersytem in der Marine genutzt hatte: Mäuse waren in der Marine nicht besonders beliebt, weil sie bei rauem Wetter immer davon liefen. Wir entschieden uns für eine Reihe von Buttons, entweder an der Seite

oder unten auf dem Bildschirmrahmen. Die Legende für Buttons besteht aus Klickbereichen in der den Buttons zugeordneten Position. Sie bekommen symbolische Namen, die sonst nicht benutzt werden.

Field	Examples	Values
		Wichtige Informationen
Type	radiobuttonshow	Kann nicht geändert werden
Title	My Live Show	Text beschreibt die Show. Wird im Editor und in Menüs angezeigt.
Show Reference	myliveshow	Eintrag, wie die Show mit anderen Shows verknüpft ist. Text ohne Leerzeichen.
Show Canvas		Nicht aktiv.
Medialist	mymedia.json	Dateiname der Medialiste, die die Tracks für die Hyperlinkshow enthält. Alle Tracks sollten eine Trackreferenz haben.
Links	name call mytrack	Siehe oben.
First Track	myfirsttrack	Die Track Referenz des Tracks, der die Einstiegsshow der Show formt.
Home Track	myhometrack	Die Track Referenz des Tracks, der den Beginn des call/return-Pfades bezeichnet.
Timeout Track		Dieser Track wird angezeigt, wenn ein Timeout geschieht. Pi Presents geht nicht sogleich zum Home-Track, weil eventuell Animationen oder konkurrierende Shows gestoppt werden sollen. Normalerweise macht dieser Track, was immer auch gefordert wird, – 'goto' zum Home-Track nach einer kurzen Zeit. Wenn es ein Audiotrack ist, kann er auf 'no sound' und eine 'zero duration' eingestellt werden.
Disable Controls	no	yes/no Wenn 'yes' dann sind interne Operationen (Start, Pause, Auf, Ab, Stopp) und Laufzeitkontrollen deaktiviert. Dies ist eine schnelle Möglichkeit, alle Steuerungen für konkurrierende Shows zu sperren. Für eine feinere Steuerung stehen die 'controls'-Felder zur Verfügung. (siehe Kapitel 6.4).
Timeout	30	Sekunden - Wenn kein Benutzerbefehl anliegt oder der Track nicht vor dem Timeout sowieso endet, erfolgt die Rückkehr zum Einstiegstrack. Ein Wert von 0 deaktiviert die Timeout-Funktion
		Show Text liegt auf allen Bildern der Show.

Field	Examples	Values
Show Text	Picture of Taj Mahal	Wenn nicht leer erscheint der Text bei allen Bildern, Videos und Audiotracks der Show.
Show Text Font	Helvetica 30 bold	Eine Schrift mit der Schriftauswahl wählen oder Schrift direkt eintragen: Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm
Show Text Colour	white	Schriftfarbe im Auswahlfenster wählen, erzeugt eine sechsstellige Hexadezimalzahl, oder Farbnamen direkt eintragen. Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm
Show Text x Position	100	Textabstand vom linken Bildschirmrand (Pixel).
Show Text y Position	100	Textabstand vom oberen Bildschirmrand (Pixel).
Controls		Values siehe Mediashow
		Tracks in der Show benötigen Einstellungen.
Background Image	+/media/image.jpg	Values siehe Mediashow
Background Colour		Values siehe Mediashow
Transition	cut	Values siehe Mediashow
Duration	10	Values siehe Mediashow
Image Window		Values siehe Mediashow
Video Player Audio	hdmi	Values siehe Mediashow
Video Player Volume	0	Values siehe Mediashow
Video Window		Values siehe Mediashow
Video Player Options		Values siehe Mediashow
Audio Player Audio	local	Values siehe Mediashow
Audio Player Speaker	stereo	Values siehe Mediashow
Audio Player Volume	0	Values siehe Mediashow
Audio Player Options		Values siehe Mediashow
Web Window		Values siehe Mediashow
Browser Options		Values siehe Mediashow

5.2.6 Start Show

Die Start Show benennt die Shows, die laufen sollen, im Start Shows-Feld. Alle Shows, die hier aufgeführt werden, werden mit Pi Presents gestartet. Jede Show läuft gleichzeitig mit den anderen, Eingabe-Aktionen werden an all diese konkurrierenden Shows geleitet.

Die Kombination der Ressourcen, die von den gleichzeitig laufenden Shows genutzt werden, sind eingeschränkt. Siehe Kapitel 6.4.4.

Start Show muss gestartet sein, sie initialisiert die ersten vom Benutzer gewählten Shows.

Field	Examples	Values
type	start	Muss immer 'start' sein, nicht veränderbar.
title	First Show	Text, der die Show beschreibt. Wird im Editor angezeigt.
Show Reference	start	muss 'start' sein.
Start Shows	show1 show2	Eine Liste der Show Referenzen, durch Leerraum getrennt.

5.3 Medialisten

Medialisten sind vergleichbar mit Playlisten eines Mediaplayers. Sie legen fest, welche Tracks abgespielt werden sollen. Wie die Tracks abgespielt werden sollen, regeln die Shows, die mit den Medialisten verbunden sind.
Jeder Eintrag in einer Medienliste ist ein Track. Tracks können unterschiedliche Typen sein:

- image – ein Einzelbild.
- video – ein Track, der vom OMXPlayer abgespielt wird.
- audio – ein Audiotrack, der vom MPplayer abgespielt wird.
- message – zeigt Textzeilen vor einem farbigen Hintergrund, ein Hintergrundbild ist möglich.
- show – Shows können als Tracks angesehen werden, die sich auf vielen Ebenen verzweigen können.
- child-show – ist die Child Show einer Mediashow oder einer Liveshow.
- menu-background – bestimmt die Bilddatei, die als Hintergrund für ein Menü benutzt wird.

Eine Medialiste ist in der Regel mit einer einzigen Show verbunden. In Pi Presents sind sie von den anderen Show-Merkmalen getrennt, so dass die gleiche Medialiste von zwei unterschiedlichen Shows benutzt werden kann.

5.4 Tracks

Jeder Track-Typ besitzt Felder, die sein Abspielen beschreiben, einige dieser Anweisungen überschreiben die entsprechenden Felder in der zugehörigen Eltern-Show.

- image – ein Einzelbild. Erlaubte Bildformate sind derzeit Bilder, die von der Python Imaging Library gerendert werden können. Die Bildgröße sollte auf etwa ein Megapixel limitiert werden.

- video – ein Track, der vom OMXPlayer abgespielt wird. Die Abspielbarkeit von Videoformaten hängt von den Codec-Lizenzen ab, die von der Foundation erworben sind.

- audio – ein Audio-Track, der vom MPlayer abgespielt wird. Spielbar ist jedes Format, dass der MPlayer kennt. Audio-Track ist sehr flexibel, weil er über ausführliche Möglichkeiten verfügt.

- message – zeigt Textzeilen vor einem farbigen Hintergrund, ein Hintergrundbild ist möglich.

Message kann zudem benutzt werden, um einen leeren Bildschirm zu erzeugen. Message-Tracks bieten einfache Folien-Funktionen an. Wenn es etwas mehr sein soll, kann man Impress (Libre Office) oder Powerpoint benutzen, die Präsentation als .jpg exportieren und als 'Image' anzeigen.

- show – Shows können als Tracks eingefügt werden. Erlaubt ist eine unbegrenzte Verzweigung in die Tiefe.

- Child-show – ist ein Track, der eine 'Child Show' einer Mediashow oder einer Liveshow repräsentiert.

- menu-background – Spezialtrack, der die Bilddatei für den Menühintergrund bezeichnet.

5.4.1 Track-Datei-Namen

Track Datei-Namen können relativ oder absolut sein. Mit relativen Dateinamen sind Profile portierbar. Siehe Kapitel 4.2.1

- Das vorangestellte + Zeichen im Dateipfad bezeichnet den Track-Speicherort relativ zum /pp_home Verzeichnis. Wenn das Pluszeichen fehlt, muss der Pfad absolut angegeben werden. Es ist ratsam, Mediadateien in /pp_home zu speichern, wenn Profile portierbar sein sollen.

- Absolute Pfade sind angebracht, wenn Internetadressen eingetragen werden sollen, z.B.: http://www.mysite.com/track_to_play.mp4

5.4.2 Unbenannte und benannte Tracks

Einige Medialisteneinträge haben Bezeichnungen, die im Feld 'Track References' beschrieben sind. Wenn 'Track Reference' leer ist, ist der Track Bestandteil eines Menüs oder einer Mediashow. Wenn das Feld nicht leer ist, gehört der Track zu einer Radiobuttonshow, einer Hyperlinkshow oder ist für einen besonderen Gebrauch bestimmt: Aktuell sind das zwei spezielle Gebrauchsmöglichkeiten:

- als Hintergrundbild für ein Menü mit der Bezeichnung pp-menu-background

- als Child Show einer Mediashow oder einer Liveshow mit der Typbezeichnung pp-child-show.

Die Felddefinitionen für jeden Typ von Medialisten-Tracks folgen im nächsten Kapitel.

5.4.3 Show Track

Die Tracks einer Show können selbst Shows sein. Sie heißen sub-shows. Showtracks erlauben das Einfügen von sub-shows in die Medialiste. Das Feld 'Show To Run' benennt die Show Referenz der Show.

5.4.4 Image Track

Image Tracks werden von der Python Imaging Library gerendert. (Siehe empfohlene Bildgröße in

den Hardwareanforderungen). Image Tracks pausieren mit der internen 'Pause'-Funktion. Bilder können in einem Fenster erscheinen, sie können ein Bild oder eine Farbe als Hintergrund benutzen, und können von einem Text überlagert sein. Text überlagert einzelne Tracks, Show Text einzelne Shows.

Field	Example	Values
Type	image	
Title		Wird in einem Menü und im Editor angezeigt.
Track Reference		Kennzeichnung eines Tracks – leer, wenn der Text innerhalb eines Menüs oder einer Mediashow läuft. Nicht leer, wenn der Track mit einem symbolischen Namen gesteuert wird oder ein spezieller Track ist.
Location	+/media/sarenam e.gif	Der Dateiname des Tracks, der auch leer sein kann.
Duration	5	Sekunden - Wenn '0' wird das Bild durch Benutzereingabe angezeigt und beendet. Wenn leer, wird der Wert in der Eltern Show benutzt.
Transition	cut	cut.
Image Window	original 10 100 fit fit 1010 40 40 fit ANTIALIAS	Ein Sichtfenster, in dem ein Bild gezeigt wird. Wenn leer, wird der Wert in der in der Eltern Show benutzt, Weiteres siehe Kapitel 5.4.4.1
Background Image	+/media/image.j pg	Der Dateiname des Bildes das als Hintergrund für das Hauptbild genutzt wird. Wenn leer, wird der Wert in der Eltern Show genutzt.
Display Show Background Image	yes	yes,no yes – das Hintergrundbild der Elternshow wird angezeigt, wenn der Track 'Background Image' leer ist. no – das Hintergrundbild der Elternshow wird nicht angezeigt.
Background Colour		Schriftfarbe im Auswahlfenster wählen, erzeugt eine sechsstellige Hexadezimalzahl, oder Farbnamen direkt eintragen. Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm Wenn leer wird der Wert der Elternshow benutzt.
Track Text	Picture of Taj Mahal	Wenn nicht leer wird der Text auf dem Bild angezeigt.
Track Text Font	Helvetica 30 bold	Eine Schrift mit der Schriftauswahl wählen oder Schrift direkt eintragen: Siehe: http://effbot.org/tkinterbook/tkinter-

Field	Example	Values
		widget-styling.htm
Track Text Colour	white	Schriftfarbe im Auswahlfenster wählen, erzeugt eine sechsstellige Hexadezimalzahl, oder Farbnamen direkt eintragen. Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm
Track Text x Position	100	Linker Textabstand vom linken Bildschirmrand (Pixel).
Track Text y Position	100	Oberer Textabstand vom oberen Bildschirmrand (Pixel).
Display Show Text	yes	yes,no Erlaubt oder sperrt die Anzeige von Showtext in diesem Track.
Links	myname play track34	Link Befehl für Radiobuttonshow und Hyperlinkshow.
Show Control at Beginning	start audio1	Siehe Kapitel 6.4
Show Control at End	stop audio1	Siehe Kapitel 6.4
Animation at Beginning	out1 on 1 out1 off 2 out2 on 1 out3 on 6	Siehe Kapitel 8
Clear Animation	no	yes/no - Siehe Kapitel 8
Animation at End	out2 off	Siehe Kapitel 8

5.4.4.1 Image Window

Image Window (Bildfenster) steuert die Positionierung eines Bildes auf der Bildschirmfläche.Es enthält eine Reihe von Feldern, die durch Leerräume getrennt sind. Wenn Image Window leer ist, wird der Wert der Show benutzt.
Das erste Feld enthält original, shrink, fit oder warp

- original
  Das Bild wird in seiner Originalgröße gezeigt. Dieser Befehl kennt zwei Argumente, die x,y, Koordinaten der oberen linken Ecke des angezeigten Bildes. Wenn die Argumente fehlen, wird das Bild zentriert auf dem Bildschirm platziert.

  z.B. original, original 100 100

- fit
  Das Bild wird im voreingestellten Fenster eingepasst und behält dabei sein Seitenverhältnis. Das Bild wird wie gewünscht geschrumpft oder gestreckt. Der Befehl verfügt über vier Argumente, die x,y, - Koordinaten des angezeigten Bildes: die obere linke und die untere rechte Ecke. Wenn keine Argumente gesetzt sind, wird das Bild auf der Bildschirmfläche zentriert.

z.B. fit, fit 100 100 1000 500, fit 100 100 1000 500 BICUBIC

- shrink
  gleich mit fit, Ausnahme: das Bild wird nicht vergrößert, wenn es kleiner als das Fenster ist.

  z.B. shrink, shrink 100 100 1000 500, shrink 100 100 1000 500 BICUBIC

- warp
  Das Bild erscheint eingepasst in das eingerichtete Fenster ohne sein Seitenverhältnis beizubehalten. Das Bild wird, wie gewünscht, geschrumpft oder vergrößert. Der Befehl verfügt über vier Argumente: die x,y, - Koordinaten der oberen linken und der unteren rechten Ecke des angezeigten Fotos. Wenn keine Argumente gesetzt sind, wird das Bild auf volle Bildschirmgröße gestreckt.

  z.B. warp, warp 100 100 1000 500, warp 100 100 1000 500 BICUBIC

Für fit, shrink und warp kann ein optionaler Filter als fünftes Argument gesetzt werden, eines von NEAREST, BILINEAR, BICUBIC, ANTIALIAS

siehe: http://effbot.org/imagingbook/image.htm
Miniaturbild und Vergrößerungs-Kapitel, wenn kein Argument gesetzt ist, wird NEAREST benutzt.

5.4.5 Video Track

Ein Video Track wird vom OMXPlayer abgespielt. Pi Presents kann jeden Track spielen, den der OMXPlayer darstellen kann. (aber: Hardware-Anforderungen beachten; siehe Kapitel 12). Videotracks können mit der internen 'Pause'-Funktion angehalten werden. Videos können in einem Fenster abgespielt werden. Sie können im Hintergrund ein Bild oder eine Farbe besitzen, beide können mit Overlay-Text belegt werden. Track-Text überlagert den einzelnen Track; Show-Text überlagert eine Show. Das Video selbst kann wegen der Beschränkungen im OMXPlayer kein Overlay tragen.

Field	Example	Values
Type	video	
Title	The Film	Wird im Menü und im Editor angezeigt.
Track Reference		Kennzeichnung des Tracks – leer, wenn der Track in einem Menü enthalten ist. Nicht leer, wenn der Track mit einem symbolischen Namen gesteuert wird oder ein spezieller Track ist.
Location	+/myvideos/film .mp4	Der gewünschte Dateiname des Tracks.
Video Player Audio	local	hdmi/local. Wenn leer, wird der Wert der Elternshow genutzt.
Video Player Volume	0	Lautstärke des Videotracks (-60 -> 0 dB). Wenn leer, wird der Wert der Elternshow genutzt.
Video Window	original	Ein Sichtfenster für das Video. Wenn leer,

	warp warp 10 10 500 1000	werden die Werte der Elternshow genutzt. • original – benutzt OMXPlayer -Standardverhalten • warp – skaliert auf Bildschirmgröße, ohne das Seitenverhältnis zu ändern. • warp – gefolgt von zwei x,y,-Paaren oben, links, unten, rechts) – des Sichtfensters. Das Video wird auf diese Größe skaliert. Das Seitenverhältnis wird nicht beibehalten.
Background Colour	red	Schriftfarbe im Auswahlfenster wählen, erzeugt eine sechsstellige Hexadezimalzahl, oder Farbnamen direkt eintragen. Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm Wenn leer, wird der Wert der Elternshow benutzt.
Background Image	+/images/back.jpg	Der Dateiname eines optionalen Bildes, dass als Hintergrund gezeigt wird, kann leer bleiben. Video, Showtext und Track werden über dem Bild angezeigt. Wenn leer, wird der Wert der Elternshow benutzt.
Display Show Background Image	yes	yes,no yes – das Hintergrundbild der Elternshow wird angezeigt, wenn der Track 'Background Image' leer ist. no – das Hintergrundbild der Elternshow wird nicht angezeigt.
Track Text	Picture of Taj Mahal	Wenn nicht leer wird der Text auf dem Bild angezeigt.
Track Text Font	Helvetica 30 bold	Eine Schrift mit der Schriftauswahl wählen oder Schrift direkt eintragen: Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm
Track Text Colour	white	Schriftfarbe im Auswahlfenster wählen, erzeugt eine sechsstellige Hexadezimalzahl, oder Farbnamen direkt eintragen. Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm
Track Text x Position	100	Linker Textabstand vom linken Bildschirmrand (Pixel).
Track Text y Position	100	Oberer Textabstand vom oberen Bildschirmrand (Pixel).

Display Show Text	yes	yes,no Erlaubt oder sperrt die Anzeige von Showtext in diesem Track.
Links	myname play track34	Link Befehl für Radiobuttonshow und Hyperlinkshow.
Show Control at Beginning	start audio1	Siehe Kapitel 6.4
how Control at End	stop audio1	Siehe Kapitel 6.4
Animation at Beginning	out1 on 1 out1 off 2 out2 on 1 out3 on 6	Siehe Kapitel 8
Clear Animation	no	yes/no siehe Kapitel 8
Animation at End	out2 off	Siehe Kapitel 8

5.4.6 Audio Track

Audiotracks werden vom MPlayer abgespielt. Pi Presents sollte alle Audiotracks spielen, die der Mplayer abspielen kann. Der Ton kann über HDMI oder analog und jeweils über einen der analogen Lautsprecher ausgegeben werden. Audiotracks können mit der internen 'Pause'-Funktion angehalten werden.

AudioTrack kann ein Bildschirm zugeordnet werden. Der Bildschirm kann eine Farbe oder ein Bild als Hintergrund besitzen. Tracktext liegt jeweils über einem Track, während Showtext über einer Show gezeigt wird.

Ein Audiotrack hat erweiterte Laufzeit- und Bildschirmsteuerungen, die es erlauben, mit einem 'Dummytrack', Animationen zu steuern oder konkurrierende Shows abzuspielen, die 'zero duration' haben und keine Medien abspielen.

Field	Example	Values
Type	audio	
Title	The Music	Wird im Menü und im Editor angezeigt.
Track Reference		Kennzeichnung des Tracks – leer, wenn der Track in einem Menü oder einer Mediashow enthalten ist. Nicht leer, wenn der Track mit einem symbolischen Namen gesteuert wird oder ein spezieller Track ist.
Location	+/tracks/music.mp3	Der Dateiname des Tracks. Die Location ist beliebig. Wenn leer, erfolgt die Zeitsteuerung im Duration Feld, das nicht leer sein darf.
Duration	5	Sekunden – wenn 'Duration' leer ist, endet der Track, wenn die Audiodatei zu Ende ist, abgebrochen wird, oder bei Abbruch durch den Benutzer, falls kein Audiotrack vorhanden ist. Wenn 'zero' dann ist die Dauer immer 'zero'.

		'Location' sollte leer sein, kein Audiotrack wird abgespielt Zero Duration wird vom Track für Show- oder Animationssteuerung benötigt. Wenn Duration größer ist als zero, endet der Track zur angegebenen Zeit. Wenn eine Datei mit Location eingerichtet wurde, kann das Abspielen des Tracks zu früh abgebrochen werden oder es gibt eine Tonpause nach der Audiodatei.
Audio Player Speaker	left	left/right/stereo – Wenn leer, wird der Wert in der Parentshow benutzt.
Audio Player Audio	local	hdmi/local - Wenn leer, wird der Wert in der Parentshow benutzt.
Audio Player Volume	0	Volume of audio track (-60 -> 0 dB) – Wenn leer, wird der Wert der Elternshow benutzt.
Clear Screen	no	yes/no Der Bildschirm wird beim Start eines Audiotracks normalerweise nicht gelöscht, weil das auch konkurrierende Anzeigen anderer Shows löschen würde. Wenn ein Audiotrack Teil einer Hauptshow ist, könnte die Löschung vorheriger Bildschirminhalte wünschenswert sein.
Background Colour	red	Schriftfarbe im Auswahlfenster wählen, erzeugt eine sechsstellige Hexadezimalzahl, oder Farbnamen direkt eintragen. Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm) Wenn leer wird der Wert in der Elternshow benutzt.
Background Image	+/images/back.jpg	Der Dateiname eines optionalen Bildes, das als Hintergrund gezeigt wird, kann leer bleiben. Video, Showtext und Track werden über dem Bild angezeigt. Wenn leer, wird der Wert der Elternshow benutzt.
Display Show Background Image	yes	yes,no yes – das Hintergrundbild der Elternshow wird angezeigt, wenn im Track 'Background Image' leer ist. no – das Hintergrundbild der Elternshow wird nicht angezeigt.
Track Text	Picture of Taj Mahal	Wenn nicht leer wird der Text auf dem Bild angezeigt.
Track Text Font	Helvetica 30 bold	Eine Schrift mit der Schriftauswahl wählen oder Schrift direkt eintragen:

		Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm
Track Text Colour	white	Schriftfarbe im Auswahlfenster wählen, erzeugt eine sechsstellige Hexadezimalzahl, oder Farbnamen direkt eintragen. Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm
Track Text x Position	100	Linker Textabstand vom linken Bildschirmrand (Pixel).
Track Text y Position	100	Oberer Textabstand vom oberen Bildschirmrand (Pixel).
Display Show Text	yes	yes,no Erlaubt oder sperrt die Anzeige von Showtext in diesem Track.
Links	myname play track34	Link Befehl für Radiobuttonshow und Hyperlinkshow.
Show Control at Beginning	start audio1	Siehe Kapitel 6.4
Show Control at End	stop audio1	Siehe Kapitel 6.4
Animation at Beginning	out1 on 1 out1 off 2 out2 on 1 out3 on 6	Siehe Kapitel 8
Clear Animation	no	yes/no Siehe Kapitel 8
Animation at End	out2 off	Siehe Kapitel 8

5.4.7 Web Track

Web Tracks werden vom webkitbasierten uzbl-Browser gerendert.

Der Browser wird mit Pi Presents installiert. Er kann, wie jedes normale Programm, vom Desktop aus benutzt werden.

Websites können aus dem Internet oder vom lokalen Dateisystem abgerufen werden. Sie können in einem Fenster oder im Vollbildmodus laufen. Als Hintergrund ist ein Bild oder eine Farbe möglich und sie können Track Text und Show Text zeigen. Track Text liegt über den einzelnen Tracks, Show Text über der Show.

Die Integration des uzbl in Pi Presents ist nicht so gelungen, wie die Zusammenarbeit mit den anderen Playern. Am besten funktioniert uzbl auf nicht interaktiven Bildschirmen:

- Der Browser öffnet sich, wenn ein Web Track gestartet wird. Das kann bis zu 15 Sekunden dauern. Für eine Reihe von Websites verfügt der Browser in Pi Presents über ein Browser-Befehlsscript, das die Darstellung einer Abfolge von Websites mit einstellbaren Abspielschleifen möglich macht.

- Im Gegensatz zu den anderen Playern fokussiert der Browser auf das Browserfenster, nicht auf das Fenster von Pi Presents. Tastatureingaben werden an uzbl gesandt und nicht an Pi Presents, bis das Pi Presents-Fenster angeklickt wird. GPIO-Eingaben sind wirkungslos.

- Das Browserfenster hat Scrollbalken und eine Titelleiste. Der Browser beeinflusst die Darstellung der Taskleiste. Kapitel 5.4.7.2 beschreibt eine Lösung.

Field	Example	Values
Type	web	
Title		Wird im Menü und im Editor angezeigt.
Track Reference		Kennzeichnung des Tracks – leer, wenn der Track in einem Menü oder einer Mediashow enthalten ist. Nicht leer, wenn der Track mit einem symbolischen Namen gesteuert wird oder ein spezieller Track ist.
Location	www.google.co.uk +/media/mywebpage.html	Der Dateiname des Tracks: • ein Internetadresse (url) • ein voller Pfadname zu einer Datei • ein relativer Pfad zu einer unter pp_home gespeicherten Datei
Duration	5	Seconds – Wenn '0' wird die Website gespielt, bis sie durch Benutzerbefehl abgebrochen wird. Wenn leer, wird der Wert in der Elternshow benutzt.
Web Window	warp warp 10 100 200 700	Ein Sichtfenster für die Webshow. Wenn leer, werden die Werte der Elternshow genutzt. • warp – skaliert auf Bildschirmgröße, ohne das Seitenverhältnis zu ändern. • warp – gefolgt von zwei x,y,-Paaren oben, links, unten, rechts) – des Sichtfensters. Die Webseite wird auf diese Größe skaliert. Das Seitenverhältnis wird nicht beibehalten.
Background Image	+/media/image.jpg	Der Dateiname eines Bildes, das als Hintergrund für die Webseite gezeigt wird. Wenn leer, wird der Wert der Elternshow benutzt.
Display Show Background Image	yes	yes,no yes – das Hintergrundbild der Elternshow wird angezeigt, wenn der Track 'Background Image' leer ist. no – das Hintergrundbild der Elternshow wird nicht angezeigt.

Field	Example	Values
Background Colour		Schriftfarbe im Auswahlfenster wählen, erzeugt eine sechsstellige Hexadezimalzahl, oder Farbnamen direkt eintragen. Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm Wenn leer, wird der Wert der Elternshow benutzt.
Track Text	Picture of Taj Mahal	Wenn nicht leer, wird der Text auf dem Bild angezeigt.
Track Text Font	Helvetica 30 bold	Eine Schrift mit der Schriftauswahl wählen oder Schrift direkt eintragen: Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm
Track Text Colour	white	Schriftfarbe im Auswahlfenster wählen, erzeugt eine sechsstellige Hexadezimalzahl, oder Farbnamen direkt eintragen. Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm
Track Text x Position	100	Linker Textabstand vom linken Bildschirmrand (Pixel).
Track Text y Position	100	Oberer Textabstand vom oberen Bildschirmrand (Pixel).
Display Show Text	yes	yes,no Erlaubt oder sperrt die Anzeige von Showtext in diesem Track.
Browser Commands	wait 10 refresh	See below
Links	myname play track34	Link Befehl für Radiobuttonshow und Hyperlinkshow.
Show Control at Beginning	start audio1	Siehe Kapitel 6.4
Show Control at End	stop audio1	Siehe Kapitel 6.4
Animation at Beginning	out1 on 1 out1 off 2 out2 on 1 out3 on 6	Siehe Kapitel 8
Clear Animation	no	yes/no – siehe Kapitel 8
Animation at End	out2 off	Siehe Kapitel 8

5.4.7.1 Browser Befehle

Das Feld für Browser Commands enthält '0' oder mehr Befehle. Jeder Befehl steht in einer eigenen Zeile. Einige Befehle haben ein Argument, durch einen Leerraum vom Befehl getrennt.

Command	**Description**	**uzbl Command used**
load <arg>	Lädt die Webseite nach Maßgabe des Arguments <arg> <arg> können sein: • eine Internetadresse (url) • ein absoluter Pfadname zu einer Datei • ein relativer Pfadname zu einer Datei, die unter pp_home gespeichert ist und ein voran gestelltes '+' besitzt	uri
refresh	Aktuell geladene Webseite neu laden	reload_ign_cache
wait <arg>	Warten <arg> Sekunden	-
exit	Eigentlich startet das Script nach dem letzten Befehl wieder neu. 'Exit' beendet aber als letzter Befehl das Abspielen des Web Tracks.	-
loop	Ein einzelner Schleifenbefehl ist erlaubt. Wenn ein Schleifenbefehl vorliegt, wird er nach dem letzten Befehl des Scriptes ausgeführt.	-
uzbl <arg>	Führt einen uzl-Browser-Befehl aus. Die Befehle sind hier beschrieben: http://www.uzbl.org/readme.php <arg> ist der uzbl-Befehl mit seinen Argumenten. Diese Befehle erlauben einen gewünschten Zuschnitt der Bildschirmdarstellung individueller Webseiten.	

Beispiel: Wiederholung von zwei Webseiten. Schleife wird fortgesetzt bis die Trackdauer erreicht ist.

```
wait 20
load www.google.co.uk
wait 20
load www.museumoftechnology.org.uk
```

Beispiel: Zeigt zwei Webseiten, wechselt dann zum nächsten Track.

```
wait 20
load www.google.co.uk
wait 20
load www.museumoftechnology.org.uk
wait 20
exit
```

Example: Lädt die geladene Webseite im 20-Sekunden-Rhythmus neu.

```
wait 20
```

refresh

5.4.7.2 Vollbild des Browser-Displays

Standardmäßig zeigt uzbl eine Titelleiste. Um sie zu entfernen, fügen Sie in die Datei /home/pi/.config/openbox/lxde-rc.xml direkt vor dem Eintrag </applications>

```
 <application name="uzbl*">
   <decor>no</decor>
 </application>
```

Erst nach einem Neustart des Pi wird diese Änderung wirksam.

Wenn das Browserfenster öffnet, wechselt das darunter liegende Pi Presents Fenster und bringt die Taskleiste zum Vorschein. Um das zu korrigieren, muss die Taskleiste fixiert und minimiert werden:

- auf die Taskbar und Panel Setting wählen
- den Auswahl-Tab klicken. Solid colour und black mit 100% Deckkraft wählen.
- Erweitert klicken
- 'minimize panel when not in use' auf On
- 'Size when minimized' auf 2 Pixel

Scrollbalken werden angezeigt, wenn der Inhalt einer Webseiten größer als das Fenster ist. Das kann hier rückgängig gemacht werden (nicht getestet):

http://www.uzbl.org/wiki/hide-scrollbars

Alternativ kann das uzbl zoom_out command in einem Browser-Befehlsscript die Seitengröße verwalten. Dann sind Scrollbalken nicht erforderlich.

5.4.8 Message Track

Message Tracks zeigen Text wahlweise vor einem farbigen Hintergrund oder einem Bild. Sie benötigen keine Mediadatei zur Einrichtung, weil der Text im Message-Textfeld enthalten ist.

Field	Example	Values
Type	message	
Title	A Message	Wird im Menü und im Editor angezeigt.
Track Reference		Kennzeichnung des Tracks – leer, wenn der Track in einem Menü oder einer Mediashow enthalten ist. Nicht leer, wenn der Track mit einem symbolischen Namen gesteuert wird oder ein spezieller Track ist.
Message Text	Welcome	Anzeigetext.

Duration	5	Seconds – Wenn '0' wird die Message gezeigt, bis sie durch Benutzerbefehl abgebrochen wird. Wenn leer, wird der Wert in der Elternshow benutzt.
Text Font	Helvetica 30 bold	Eine Schrift mit der Schriftauswahl wählen oder Schrift direkt eintragen: Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm
Text Colour	white	Schriftfarbe im Auswahlfenster wählen, erzeugt eine sechsstellige Hexadezimalzahl, oder Farbnamen direkt eintragen. Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm
Justification	left	left/center/right – Textposition.
Message x Position	100	Wenn leer, wird die Message auf dem Bildschirm zentriert. Wenn nicht leer, der Abstand des linken Textendes vom linken Bildschirmrand (Pixel).
Message y Position	500	Wenn die Message-x- Position angegeben ist, der Abstand vom oberen Textende zum oberen Bildschirmrand (Pixel).
Background Colour	red	Schriftfarbe im Auswahlfenster wählen, erzeugt eine sechsstellige Hexadezimalzahl, oder Farbnamen direkt eintragen. Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm Wenn leer, wird der Wert der Elternshow benutzt.
Background Image	+/images/back.jpg	Dateiname eines Bildes, das an Stelle eines einfachen Hintergrundes angezeigt wird. Die Message wird am oberen Bildrand angezeigt. Wenn leer, wird der Wert der Elternshow benutzt.
Display Show Background Image	yes	yes,no yes – das Hintergrundbild der Elternshow wird angezeigt, wenn das Trackfeld 'Background Image' leer ist. no – Das Hintergrundbild der Elternshow wird nicht angezeigt.
Track Text	Picture of Taj Mahal	Wenn nicht leer, wird dieser Text auf dem Bild angezeigt.
Track Text Font	Helvetica 30 bold	Eine Schrift mit der Schriftauswahl wählen oder Schrift direkt eintragen: Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm

Track Text Colour	white	Schriftfarbe im Auswahlfenster wählen, erzeugt eine sechsstellige Hexadezimalzahl, oder Farbnamen direkt eintragen. Siehe: http://effbot.org/tkinterbook/tkinter-widget-styling.htm
Track Text x Position	100	Linker Textabstand vom linken Bildschirmrand (Pixel).
Track Text y Position	100	Oberer Textabstand vom oberen Bildschirmrand (Pixel).
Display Show Text	yes	yes, no Erlaubt oder sperrt die Anzeige von Showtext in diesem Track.
Links	myname play track34	Link Befehl für Radiobuttonshow und Hyperlinkshow.
Show Control at Beginning	start audio1	Siehe Kapitel 6.4
Show Control at End	stop audio1	Siehe Kapitel 6.4
Animation at Beginning	out1 on 1 out1 off 2 out2 on 1 out3 on 6	Siehe Kapitel 8
Clear Animation	no	yes/no – Siehe Kapitel 8
Animation at End	out2 off	Siehe Kapitel 8

5.4.9 Show Track

Eine Show kann ein Track in einer anderen Show sein. Möglich sind:

- Eine Mediashow , zusammengestellt aus kleineren Mediashows
- Ein Menü von Mediashows, zumeist Präsentationen
- Menüs mit Untermenüs
- Eine Liveshow gestartet von einer Mediashow, die den Startbildschirm liefert

Field	Example	Values
Type	show	
Title	My Other Show	Wird im Menü und im Editor angezeigt.
Track Reference		Kennzeichnung des Tracks – leer, wenn der Track in einem Menü oder einer Mediashow enthalten ist. Nicht leer, wenn der Track mit einem symbolischen Namen gesteuert wird oder ein spezieller Track ist.
Show to Run	myothershow	Verknüpfung zur Show, die laufen soll.

5.4.10 Child Show Track

This is a labelled show entry which provides a referencing mechanism for the child show of a mediashow.

Field	Example	Values
Type	show	
Title	Child Show	Wird im Menü und im Editor angezeigt.
Track Reference	pp-child-show	Track hat eine Verknüpfung zu einer Child-Show. Der Track ist nicht Bestandteil im Menü oder der Mediashow.
Show to Run	mychildshow	Verknüpfung zur Show, die laufen soll.

6 Shows steuern

6.1 Einführung

Die Show Control in Pi Presents ist sehr flexibel und erlaubt die Zusammenstellung komplexer Anwendungen. Für die meisten Benutzer wird der Auslieferungsstandard aber sicher ausreichen.

Es gibt eine Reihe von Steuerungstechniken für Shows und ihre Tracks, jede hat eine spezielle Aufgabe:

- Internal Operations - play, pause, stop, up und down - Internal Operations dienen der Steuerung von Mediashows, Menüs und Liveshows

- Triggers - Triggers (Auslösepunkte) werden für Mediashows benutzt, als Alternative zu Internal Operations. Sie werden in erster Linie mit GPIO-Eingaben benutzt

- Links - Links werden von Radiobuttonshows und Hyperlinkshows genutzt, um zwischen Tracks zu wechseln

- Run-time Controls - OMXPlayer und MPlayer ermöglichen die Steuerung einiger Abspielfunktionen, während ein Track läuft

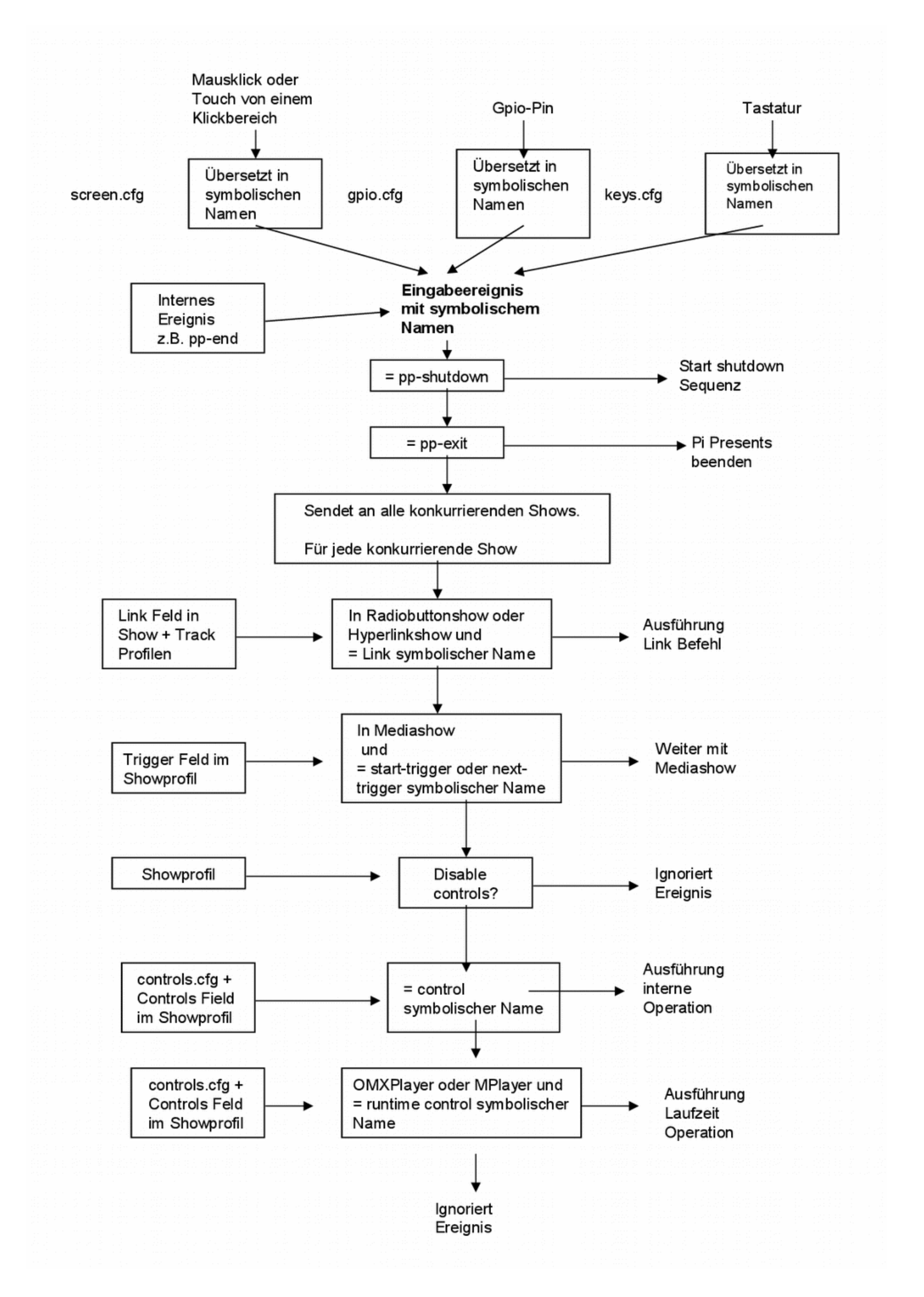
Alle Steuerungen in Pi Presents - Internal Operation, Trigger, Link or Run-time control - sind mit einem symbolischen Namen verbunden. Standardmäßig ist zum Beispiel die 'up' Internal Operation in der Datei controls.cfg an den symbolischen Namen pp_up gebunden. Außerdem möchte man eventuell in einer Radiobuttonshow einen Befehl zum Abspielen eines Tracks einfügen mit

myname play mytrack

im Links-Feld eines Tracks sind 'myname' and 'pp-up' symbolische Namen.

Reale Aktionen wie eineGPIO-Schaltung, ein Tastendruck oder ein Mausklick werden von Pi Presents in ein 'input-event' verwandelt, das über einen symbolischen Namen identifiziert wird.

Dieser symbolische Name eines 'input events' wird mit den symbolischen Namen der controls abgeglichen um die gewünschte Aktion zu bestimmen. Dieser Vergleich funktioniert, wie unten beschrieben.


Mausklick oder Touch von einem Klickbereich
Gpio-Pin
Tastatur
screen.cfg
Übersetzt in symbolischen Namen
gpio.cfg
Übersetzt in symbolischen Namen
keys.cfg
Übersetzt in symbolischen Namen
Internes Ereignis z.B. pp-end
Eingabeereignis mit symbolischem Namen
= pp-shutdown
Start shutdown Sequenz
= pp-exit
Pi Presents beenden
Sendet an alle konkurrierenden Shows.
Für jede konkurrierende Show
Link Feld in Show + Track Profilen
In Radiobuttonshow oder Hyperlinkshow und = Link symbolischer Name
Ausführung Link Befehl
Trigger Feld im Showprofil
In Mediashow und = start-trigger oder next-trigger symbolischer Name
Weiter mit Mediashow
Showprofil
Disable controls?
Ignoriert Ereignis
controls.cfg + Controls Field im Showprofil
= control symbolischer Name
Ausführung interne Operation
controls.cfg + Controls Feld im Showprofil
OMXPlayer oder MPlayer und = runtime control symbolischer Name
Ausführung Laufzeit Operation
Ignoriert Ereignis

Ausführung von 'Input Events'

Wie oben beschrieben: eine Show kann Subshows und Tracks enthalten. 'Input events' werden komplett durchgereicht, so erreichen sie die aktuell niedrigste Komponenten-Ebene.

Wenn die Konfiguration (.cfg) bearbeitet wird, müssen symbolische Namen vergeben werden. Es wird empfohlen, keine Namen beginnend mit 'pp-' einzurichten, um Konflikte mit in Pi Presents vorhandenen Namen zu vermeiden.

6.2 Steuerung von Menüs, Mediashows and Liveshows

Die Steuerung von Menüs, Mediashows und Liveshows erfolgt über 'Internal Operations'. Standardmäßig sind diese Operationen an einen Satz symbolischer Namen gebunden und diese wiederum sind mit einem frei wählbaren Set von Tasten auf einer Tastatur und GPIO-Eingaben verknüpft:

Internal Operation	**'Out of the box'** GPIO Pin siehe Kapitel 9			**Effect**
	Symbolic Name	Key	GPIO Pin.	
up	pp-up	Cursor Up		Vorheriger Track in der Mediashow oder 'up' zum Menü
down	pp-down	Cursor Down		Nächster Track in der Show oder 'down' zum menu
play	pp-play	Return		Startet einen Menüeintrag oder eine Childshow
pause	pp-pause	Spacebar		Schaltet auf Pause in Tracks die 'pause' unterstützen
stop	pp-stop	Escape		Stoppt Abspielen eines Tracks oder zu einer Elternshow zurück

Die Standardtaste für jede Operation ist in der keys.cfg, und für die GPIO-Pins in der gpio.cfg gesetzt. Die Verbindungen können modifiziert werden, wie in Kapitel 10.2 beschrieben. Die Verbindung interner Operationen mit symbolischen Namen erfolgt in controls.cfg. Es ist unwahrscheinlich, dass diese Einstellungen verändert werden müssen, es könnte aber sein, das sie überschrieben werden sollen, um eine individuelle Show zu erstellen, wie es nachfolgend beschrieben ist.

Denken Sie daran, wenn mehr als eine Show konkurrierend läuft, wird das Input-Event durchgeschleift und möglicherweise auch von jeder anderen, gleichzeitigen Show ausgeführt. Das kann unerwünscht sein, wenn zum Beispiel eine Mediashow mit Audiotracks als Hintergrund für eine manuell fortgesetzte Diashow läuft. Weil die Audiotracks nicht auch durch 'up'- und 'down'-Operationen wechseln sollen, benutzen wir das Feld Disable Controls in der Audioshow.

Wenn Disable Controls = yes, dann muss die Show automatisch laufen, oder so gesteuert werden, wie es im nächsten Kapitel geschildert wird.

Für höhere Anforderungen können individuelle Operationen begrenzt, oder mit alternativen symbolischen Namen verbunden werden um eine besondere Show zu erstellen, in dem die Steuerungsfelder einer Show verändert werden. Das wird im Kapitel 10.2.4 genauer beschrieben.

6.2.1 Mediashows and Liveshows steuern

Mediashows lassen sich gut steuern, Liveshows im begrenzten Umfang auch. Die Hinweise für die Mediashow in den folgenden Absätzen sind auch für Liveshows gedacht. Die Fähigkeiten sind für das Abspielen der Show ohne Benutzer-Interaktion eingerichtet; wenn die internen Operationen aber nicht ausgeschaltet sind, können sie auch für die Steuerung der Shows genutzt werden (mit Vorsicht!).

Theoretisch können Steuerungen den Subshows und Childshows hinzugefügt werden. Das habe ich aber nicht ausführlich getestet.

6.2.1 Start steuern

So wie die Mediashow läuft, fragt sie im Feld Trigger For Start, was als nächstes zu tun ist, auch wenn nach jedem Track Trigger For Next beauftragt werden kann, die Bewegung des nächsten Tracks zu steuern.

Trigger For Start kann folgende Werte annehmen:

- start
  Die Mediashow setzt sich bis zum ersten Track fort.

- input/input-quiet
  Die Mediashow wartet auf eine Eingabe, bevor sie den ersten Track in der Show startet. Der symbolische Name der Eingabe muss im Trigger Input Feld stehen.

  Wenn eine Eingabe per GPIO erfolgt ist, identifiziert Pi Presents den symbolischen Namen und führt die Eingabe-Aktion aus. Symbolische Namen sind den GPIO-Pins in der gpio.cfg zugeordnet (siehe Kapitel 10.2.1). Standardmäßig ist P1-11 als Eingang konfiguriert und dem symbolischen Namen PIR zugeordnet. Damit ist die Kontinuität mit früheren Versionen von Pi Presents gewährleistet. Wenn also PIR im Trigger Input Feld eingetragen ist, startet die Show, wenn P1-11 auf '0' wechselt

  Es gibt ein ungewolltes Feature in Pi Presents; eine wartende Show wird auch durch die 'Play' und 'Down' Funktionen gesteuert. Wenn nur diese Funktionen genutzt werden sollen, wählen sie 'input' oder 'input-quiet' und lassen das Trigger Input Feld leer.

- time/time-quiet

  Um die day of time (Tageszeitfunktion) zu nutzen, muss der Pi im Internet sein, oder sie

investieren in eine Hardware mit real time clock als Add-on.

Die Mediashow wartet auf eine 'time of day', bevor sie fortfährt. Im Trigger Input Feld muss eine Liste von 'times of day' enthalten sein.

Jede 'time of day' kann in Stunde, Minute und bei Bedarf Sekunde im 24-Stundenformat angegeben werden, jeweils durch Leerraum getrennt. Die Einträge dürfen in beliebiger Reihenfolge sein, z.B.:

21:2 21:03 9:30 9:30:30 0:0 23:59:59

Pi Presents beginnt mit der Show, wenn die Systemzeit eine der Zeiten in der Liste erreicht hat. Wenn eine Show zu diesem Zeitpunkt bereits läuft, wird sie nicht unterbrochen oder es wird an eine anstehende Show erinnert.

Wenn in der Liste von heute keine Shows mehr vorhanden sind, wird die erste Show in der Liste eingeplant und morgen gestartet.

Wenn die Option 'time' lautet und nicht 'time-quiet' erscheint eine Admin-Nachricht während des Wartens auf den Showstart. Die Admin-Nachricht kann die Zeit der nächsten Show angeben, siehe Kapitel 10.3 Dort gibt es einen Alternativtext für Shows, die morgen starten.

Pi Presents könnte Problem mit dem Wechsel zur Sommerzeit oder anderen nonlinearen Änderungen der Ortszeit/Systemzeit haben.

Es gibt eine Alternative zu den Angaben von 'time of day'. Wenn +10+40 statt beispielsweise 09:30 angegeben, dann wird die Show in 10 Sekunden und dann in 40 Sekunden nach dem Start der Mediashow abgespielt. Das ist aber nur für Testzwecke sinnvoll.

6.2.1.2 Trigger For End

Einige Mediashows, die sich wiederholen, oder deren Sequence=shuffle, so wie alle Liveshows enden nicht von selbst. Sie müssen mit sanfter Gewalt beendet werden.

Das kann unter bestimmten Umständen sinnvoll sein, z.B.:

- Eine den ganzen Tag laufende Liveshow soll zu einer bestimmten Tageszeit von einer 'Wir schließen jetzt'-Show abgelöste werden. Die beiden Shows müssen gleichzeitig laufen. Die Liveshow wird mit einem zeitbasierten Trigger For End' terminiert. Die 'Wir schließen jetzt'-Show bekommt einen zeitbasierten 'Trigger For Start' um kurz nach dem Ende der Liveshow weiter zu machen.
- Beenden einer sich wiederholenden Mediashow oder einer Mediashow mit der Sequence=shuffle. Letztere kann nicht von selbst enden, auch wenn es eine One Shot Show ist, denn: Wo ist das Ende einer Zufallsequenz? In diesem Fall sollte man die 'Duration'-Option der 'time of day'-Option vorziehen.

'Trigger for End' kann folgende Werte annehmen:

- none

'The end trigger' ist nicht eingeschaltet.

- time
  Um 'time of day' nutzen zu können, muss der Pi im Internet sein, oder Sie investieren in eine Hardware mit real time clock Add-on.

  Eine Mediashow läuft bis zur gewählten Tageszeit/time of day. Im 'Trigger End Times'-Feld muss eine Liste der gewünschten Tageszeiten aufgeführt sein.

  Jede Tageszeit kann in Stunden, Minuten und bei Bedarf in Sekunden im 24 Stundenformat, jeweils getrennt durch Leeraum, angegeben werden.z.B.:

  21:2 21:03 9:30 9:30:30 0:0 23:59:59

  Pi Presents beendet die Show, wenn die Orts-/Systemzeit die in der Liste vorgesehene Zeit erreicht hat.

  Das System erlaubt das Abspielen von Shows über Mitternacht hinaus. Eine um 23.55 gestartete Show kann um 00:05 am nächsten Tag beendet werden.

  Pi Presents könnte Probleme mit dem Wechsel der Sommerzeit und anderen, nonlinearen Zeitwechseln der Orts-/Systemzeit haben.

  Es gibt eine Alternative zu den Angaben von 'time of day'. Wenn +20+50 statt beispielsweise 09:30 angegeben, dann wird die Show in 20 Sekunden und dann in 50 Sekunden nach dem Start der Mediashow abgespielt. Das ist aber nur für Testzwecke sinnvoll.

- duration
  Die Mediashow endet nach einer bestimmten Zeitdauer. Der Zeitraum sollte im 'Trigger End Time'-Feld als Stunde, Minute, Sekunde eingetragen werden.

  Das Format ist h:m:s Stunden und Minuten sind optional. z.B.:

  5 - 5 seconds
  6:0 - 6 minutes
  10:0:0 - 10 hours

6.2.1.3 Trigger For Next

Das 'Trigger For Next'-Feld ermöglicht, das einzelne Tracks einer Mediashow durch eine Eingabe gesteuert werden können. Für das korrekte Funktionieren sollte 'Progress' auf 'manual' gestellt werden:

- none
  Die Steuerung ist abgeschaltet.

- input
  Die Show geht zum nächsten Track, wenn sie eine Eingabe mit dem symbolischen Namen in 'Next Input' erhält.

6.3 Radiobuttonshows und Hyperlinkshows steuern

Diese Shows werden mit Befehlen gesteuert, die in ihren Link Feldern und den Link Feldern ihrer Tracks stehen. Das wird in Kapitel 5.2.5 und Kapitel 5.2.4 genauer beschrieben. Zum Beispiel spielt der Befehl im Links Feld einer Radiobuttonshow

myname play mytrack

einen Track mit der Track Referenz 'mytrack', wenn eine Eingabe mit dem symbolischen Namen 'myname' erfolgt. Links sind nicht durch 'Disable Controls' abgeschaltet,

Radiobuttonshows und Hyperlinkshows benutzen keine internen Operationen, doch einige ihrer Tracks und Shows könnten sie benutzen.

6.4 Konkurrierende Shows steuern

6.4.1 Einführung

Pi Presents kann zwei oder mehrere Shows gleichzeitig laufen lassen. Diese konkurrierenden Shows beginnen gleichzeitig zu laufen und Eingaben gehen an alle konkurrierenden Shows. Das ist korrekt, aber wenn es nicht vorsichtig gesteuert wird, drohen unerwünschte Effekte. Die meisten normalen Situationen können jedoch mit Disable Controls gemeistert werden

- Einen Audiotrack als Hintergrund für eine Diashow nutzen.
  Richten Sie zwei Mediashows ein, eine mit einer manuell gesteuerten Slideshow und die andere mit den Audiotracks. Letztere verlangt das Abschalten der Steuerung in Disable Controls, wenn dort irgend eine Benutzer-Interaktion mit Ersterem eingetragen ist.

- Shows zu einer bestimmten Tageszeit laufen lassen.
  Richten Sie zwei oder mehr Liveshows oder Mediashows ein, jede zu einer bestimmten Tageszeit abrufbar. Sie dürfen sich nicht überlappen. Liveshow und Mediashow können auch zu bestimmten Tageszeiten oder nach Zeitperioden gestoppt werden.

- Wenn man richtig geizig ist kann man zwei vollkommen unterschiedliche Aufgaben mit einem Pi erledigen, vielleicht eine Diashow an der Rezeption mit Child Show-Fähigkeiten und einem Dummy, der in einem Ausstellungstrakt etwas erzählt, gesteuert von einer PIR innerhalb einer Single Shot Mediaswow. In diesem Fall schaltet man die Steuerung für die Ausstellung ab und verlässt sich auf das Trigger For Start Feld, das nicht von Disable Controls beeinträchtigt wird.

Komplexere Szenarios können angegangen werden, indem man die Anbindung der Controls für besondere Shows überschreibt. Die Standard-Anbindungen von symbolischen Namen und internen Operationen finden sich in der controls.cfg. Im Controls Feld einer Show kann eine interne Operation an einen anderen symbolischen Namen für die Show angebunden und deswegen mit einem anderen Input gesteuert werden. Alternativ kann man ein Überschreiben einzelner Operationen für die Show abschalten.

Mehr Ausführungen dazu in Kapitel 10.2.4

6.4.2 Shows starten und stoppen

Konkurrierende Shows laufen parallel. Es gibt zwei Methoden, um konkurrierende Shows zu starten und zu stoppen:

- Starten Sie eine oder mehrere Shows, indem Sie ihre Showreferenzen in das Start Shows Feld der Start Show eintragen. Alle Shows, die im Start Shows Feld der Startshow enthalten sind, werden laufen, wenn Pi Presents startet.

- Benutzen sie einen Control Befehl im Show Control Feld eines Tracks, um eine Show zu starten oder zu stoppen z.B.:

  myshow start
  myothershow stop

Shows, die in der Startshow gestartet wurden, können mit einem Befehl in Show Control gestoppt werden.

Nur eine Instanz einer Show kann zu einer Zeit laufen. Versuche, eine Show zu starten, die bereits läuft, werden ignoriert.

6.4.3 Pi Presents verlassen und den Pi herunterfahren

Im Zusammenhang mit Starten und Stoppen von Shows können Show Control Befehle genutzt werden, um Pi Presents zu verlassen oder den Raspberry Pi herunterzufahren, z.B.:

pipresents exit
gobbledegook shutdownnow

Jedes Wort kann an Stelle des Wortes Pi Presents benutzt werden.

6.4.4 Einschränkungen bei konkurrierenden Shows

Konkurrierende Shows haben einige Einschränkungen, die durch Linux, Mplayer und OMXPlayer bedingt sind:

- In der Regel können zwei Shows nicht gleichzeitig auf dem Bildschirm erscheinen.

- Wenn sich überlagernde Audiotracks vom MPlayer abgespielt werden, sollten sie den gleichen Ausgang benutzen, entweder HDMI oder analog. Wenn nicht, gibt es ein Umschalten auf der Hälfte der Tracks und dabei stürzte mir der Pi ab. Wenn überlagernde Audiotracks abgespielt werden sollen, kann man für einen der beiden Tracks den Videoplayer einsetzen, oder die Tracks auf den linken und rechten Audioausgang verteilen.

- HD-Videos könnten ruckeln, wenn ein Audiotrack abgespielt wird.

- Animationen starten verspätet, wenn sie gleichzeitig mit einem Audiotrack oder einem Bild gestartet werden. Das Output-Ereignis verschwindet nicht, sondern kommt lediglich etwas später.

7 Fernsteuerung

Pi Presents war eigentlich nicht für Fernsteuerungen vorgesehen, jedoch habe ich auf vielseitigen Wunsch die Fähigkeit eingebaut, dass eine Liveshow Tracks abspielen kann, die dynamisch von einem anderen Computer per FTP zugespielt werden.

Die New> Liveshow Vorlage ist eine laufende Liveshow. Die zu spielenden Tracks müssen im Verzeichnis /home/pi/pp_home/pp_live_tracks gespeichert sein.

Die von der Liveshow erkannten Tracktypen sind in den ersten Zeilen von pp_definitions.py aufgeführt.
Tracks können mit Filezilla (oder ähnlichen Tools) auf den Pi übertragen werden. Mit dem Kommandozeilenparameter -l kann man einen zweiten Speicherort für Livetracks bestimmen: Zum Ersten im Daten-Heimatverzeichnis (siehe oben) und zum Zweiten in einem Verzeichnis, das mit der -l Option festgelegt wird. Dieses Verzeichnis könnte auf einem Remote Server liegen (Ich habe das nicht getestet, Samba erforderlich?).

Alternativ könnte das komplette Showprofil auf einem Remote Fileserver liegen.

Der Pi sollte feste IP-Adressen bekommen und außerdem ist Netzwerkschutz angesagt. Mindestens das Passwort für den User Pi sollte geändert werden.

8 Animationen steuern

Für alle Tracktypen können Animationen gesteuert werden. Mit den Befehlen in 'Animation at Beginning' und 'Animation at End' Feldern kann eine GPIO-Ausgabe ein- oder ausgeschaltet werden und dabei mit dem Start oder Ende eines Tracks synchronisiert werden. Das ist auch zeitversetzt möglich.

Hier ein Beispiel für Animationsbefehle:

animate at beginning	out1 on 1 out1 off 2 out2 on 1 out3 on 6
Clear Animation	no
Animation at End	out2 off

Ein Befehl hat drei durch Leerraum getrennte Felder und wird mit einer neuen Zeile beendet:

- Symbolic Name – Name der Ausgabe, wie er in der gpio.cfg festgelegt ist (Kapitel 10.2.1). Für jedes Profil kann es unterschiedliche gpio.cfg geben.

- State to go to - on or off
- Delay – Sekunden als ganze Zahl oder 0. Wenn das Feld leer ist, wird der Wert 0 angenommen.

Der Status von GPIO-Ausgaben vor dem Pi Presents Start ist nicht immer klar und scheint bei den Pins je nach Version des Pi unterschiedlich. Also Vorsicht beim Anschluss an Output-Pins.

Alle Ausgänge werden beim Start von Pi Presents abgeschaltet. Befehle in Animation at Beginning werden mit Beginn eines Tracks und Animation at End am Ende eines Tracks ausgeführt. Wenn die Befehle ausgeführt werden, stehen die gewünschten Änderungen an den GPIO-Pins in einer Warteschlange, um zur vorbestimmten Zeit „abgefeuert“ zu werden. Trackpausen haben keinen Einfluss auf die Animationszeiten.

Animationsbefehle in der Warteschlange sind nicht verloren, wenn der Track zuende ist. So können Animationen über eine Show hinaus laufen. Ein Nebeneffekt: Ein GPIO-Wechsel kann noch erfolgen, wenn ein Track oder eine Show bereits beendet sind. Wenn das aber vermieden werden soll, und man sicher sein möchte, dass die Ausgänge mit Ende eines Tracks in einem bestimmten Zustand sind, dann muss 'yes' Animate Clear gesetzt werden. Wenn 'yes' gilt, wird die Warteschlange der Ereignisse gelöscht, die zwar vom Track zum Abspielen vorgesehen waren, jedoch nicht zum Zuge kamen – bevor Animation at End ausgeführt wird.

9 Black Box Operation

Es gibt eine Menge Einstellungsmöglichkeiten mit denen man Pi Presents zu einem bildschirmfüllenden, automatisch startenden, von GPIO gesteuerten Programm machen kann. Man benötigt aber nicht alle.

9.1 Optionen auf der Befehlszeile

```
python pipresents.py -h
```

zeigt alle Befehlszeilenkommandos

Options	
-p --profile <arg>	Name des benutzten Profils, z.B. pp_mediashow Wenn das nicht ausgewählt ist, lautet der Standard: pp_profile.
-g --gpio	Macht das Pi-GPIO-System verfügbar. Dafür muss Pi Presents als Root ausgeführt werden. `sudo python pipresents.py -g`
-b --noblank	Verhindert Bildschirmpause. Damit es funktioniert, müssen die x11-server-utils installiert sein.
-f --fullscreen	Pi Presents läuft im Vollbildmodus
-o --home <arg>	Adresse des Pi Presents 'data home' Verzeichnisses z.B. /media/USBSTICK oder /home/pi/my_data Wenn diese Option nicht gewählt ist das User Heimatverzeichnis Standard **Achtung**: Wenn sudo benutzt wird lautet das User-Heimatverzeichnis /root. Die Auswahl muss mit einem absoluten Pfad

	getroffen werden. Um ganz sicher zu sein, sollte man den absoluten Pfad angeben, wenn pipresents.py von außerhalb des pipresents-Verzeichnisses ausgeführt wird.
-d --debug	Laufzeitfehler produzieren eine Meldung und werden im Terminalfenster angezeigt, wenn es geöffnet ist. Ein eingestelltes Debugging liefert zusätzlich eine Liste der Pi Presents Operationen im Terminalfenster. Fehler und das Protokoll sind zudem in der /pipresents/pp_log.log Datei festgehalten.
-v --validate	Prüft das Profil, wenn Pi Presents gestartet wird.
-l --liveshow <arg>	Liveshows werden immer vom Verzeichnis pp_live_tracks im Datei-Heimatverzeichnis „abgeholt“. Standardmäßig verweist die Option -o darauf: `/home/pi/pp_home/pp_live_tracks` Wenn die -l Option einen Verzeichnispfad enthält, ist das eine zweite Quelle für Liveshowtracks. z.B. `/home/pi/mylivetracks`

9.2 Profil auswählen

Die --profile (-p) Befehlszeilenoption legt das zu verwendende Profil fest. Dies ist der Name des Profilverzeichnisses im Ordner 'pp_profiles'. Wenn die Option nicht gewählt ist gilt 'pp_profile' als Standard.
Das Präfix 'pp_' ist lediglich eine Kennzeichnung des Entwicklers und bedeutet nichts für den Programmablauf.

9.3 Heimatverzeichnis auswählen

Die --home (-o) Befehlszeilenoption gibt das Pi Presents Homeverzeichnis an. Wenn diese Option nicht gewählt ist, gilt standardmäßig das Heimatverzeichnis des Benutzers. **Achtung:** Wenn sudo benutzt wird, ist das Benutzer-Homeverzeichnis /root und die Option muss entsprechend erweitert werden. Um ganz sicher zu gehen, sollte der absolute Pfad gewählt werden, wenn pipresents.py von einem anderen Ort als dem pipresents-Verzeichnis ausgeführt wird.

USB-Sticks können zur Speicherung von Medien und Profilen dienen. Sie werden als Mountpoints im /media Verzeichnis aufgeführt:

z.B. `/media/KINGSTON`

Raspbian mounted USB-Sticks automatisch beim Hochfahren, erkennt sie aber auch danach. Wenn Pi Presents mit dem Einschalten des Pi gestartet wird, dauert es etwa 10 Sekunden, bis der Stick gemountet ist.

Raspbian generiert den Namen des Mountpoints aus dem Namen des Laufwerks auf dem Stick. Wenn der Stick auf einem Windowsrechner vorbereitet wurde hat Windows alle Laufwerksnamen in

Großbuchstaben konvertiert.

9.4 Pi Presents mit GPIO steuern

Die GPIO-Steuerung wird mit der --gpio Befehlsoption aktiviert. Wenn GPIO benutzt wird, muss dem Startkommando für Pi Presents 'sudo' vorangestellt werden.

Bei Auslieferung sind die unten aufgeführten Pins des GPIO Anschlusses 1 mit den ebenfalls aufgeführten symbolischen Namen in der gpio.cfg verbunden. Diese Verbindungen können, wie in Kapitel 10.2.1 beschrieben, geändert werden.

Bound Pin	Symbolic Name	Bound Key	Function
Internal Operations			
P1-15	pp-down	Cursor Down	Next in der Show oder down im Menü
P1-16	pp-up	Cursor Up	Previous in der Show oder up im menu
P1-18	pp-play	Return	Start einer Mediashow oder Abspielen der Standard Childshow
P1-22	pp-pause	Spacebar	Pausenschalter für Videos.
P1-7	pp-stop	Escape	Stopp eines laufenden Track oder Rückkehr zur Show auf der höheren Ebene
User Binding			
P1-11	PIR	-	Keine interne Operation
Special Bindings			
P1-12	pp-shutdown	-	Für 5 Sekunden drücken um Pi Presents zu beenden und den Pi herunter zu fahren
-	-	CTRL-BREAK	Abbruch von Pi Presents. Fokus in einem Pi Presents Fenster

Die GPIO-Ports sind als über ansteigende Flanke gesteuerte Eingabeports mit internen Pull-up Widerständen und für folgende Gerätecharakteristik ausgelegt:

- Drucktasten sollten mechanisch sein, 'drücken für Aktion' (normalerweise offen) und an einen GPIO-Pin und Erde (ground) angeschlossen werden.
- PIRs haben regulär geschlossene Relaiskontakte und werden zwischen GPIO-Pin und Erde (ground) angeschlossen.
- Eingaben können von normal offen auf normal geschlossen und umgekehrt geändert werden, wenn die Ansteuerung, wie sie in Kapitel 10.2.1 beschrieben ist, geändert wird.

Kapitel 10.2.1

330 Ohm Widerstände in Serie mit den Tasten oder PIR werden empfohlen, um den Raspberry Pi zu schützen, falls Eingänge versehentlich als Ausgänge verwendet werden.

GPIO Pin-----330 Ohm-----Kontakt-----0 Volt (Gnd)

Achten Sie darauf, dass kein GPIO-Pin mit dem +5-Volt-Pin in Kontakt kommt – das wäre, als würden Sie den Pi grillen.
Pi Presents enthält eine softwareseitige Entprellung mit geringer Schalthysterese. Wenn es Probleme mit Kontaktprellung gibt, kann man den Schwellwert für den betreffenden Pin in der gpio.cfg in 'pp_buttons.py' verändern (siehe Kapitel 10.2.1).

9.5 Bildschirmabschaltung deaktivieren

Um 'Bildschirm abschalten' zu deaktivieren, muss xset installiert sein, xset ist Teil des x server utilities package und sollte mit der Erstinstallation vorhanden sein. Ansonsten gilt:

```
sudo apt-get install x11-xserver-utils
```

Sie können 'Bildschirm abschalten' mit der --noblank Befehlsoption deaktivieren.

9.6 Pi Presents mit dem Raspberry Pi einschalten

Das funktioniert nur, wenn in der raspi.config 'boot to dektop' eingestellt wurde.

- Richten Sie den Ordner /home/pi/.config/lxsession/LXDE ein
  Achtung: Das Verzeichnis .config ist bereits vorhanden. Um es zu sehen, müssen Sie 'Show Hidden Files' im Dateimanager wählen.
  Achtung: Ab Januar 2015 neuer Ordner für die Autostartdatei: /home/pi/.config/lxsession/LXDE-pi

- In diesem Ordner richten Sie eine Datei 'autostart' ein, die in einer Zeile den absoluten Pfad zu pipresents und dem Homeverzeichnis enthalten soll:
  z.B.
  `sudo python /home/pi/pipresents/pipresents.py -o /home/pi -p myprofile`

- Machen Sie die Datei ausführbar, indem Sie im Dateimanager die Rechte an der Datei ändern.

9.7 Raspberry Pi mit GPIO herunterfahren

Drücken Sie den Shutdown-Button für etwa 5 Sekunden. Pi Presents beendet sich und der Pi fährt sicher herunter.

Der Pi kann kann auch mit einem Kommando in Show Control heruntergefahren werden (siehe Kapitel 6.4.3).

10 Pi Presents Einrichten

10.1 Speicherort der Konfigurationsdateien

Die Auslieferungs-Konfiguration von Pi Presents ist in mehreren Dateien in /pipresents/pp_home pipresents festgelegt:

- gpio.cfg - gpio inputs und animation outputs
- keys.cfg – keyboard (Tatstatur)
- screen.cfg - Klickbereiche auf dem Monitor)
- controls.cfg - Interne und Laufzeit-Operationen
- resources.cfg – Admin-Hinweise (Messages)

Die Konfigurationsdateien sind Textdateien, die mit dem Editor Leafpad editiert werden können. Bitte die Dateien nicht innerhalb von /pipresents/pp_home speichern; wenn sie geändert werden sollen, kopieren sie die Dateien entweder:

- innerhalb eines Profils, z.B. /home/pi/pp_home/pp_profiles/myprofile
- im Daten-Heimatverzeichnis, z.B. /home/pi/pp_home

Pi Presents sucht .cfg Dateien zuerst in einem Profil, dann im Daten-Heimatverzeichnis und erst danach in /pipresents/pp_home.

Beachten Sie beim Ändern dieser Dateien:

- die Dateien in /pipresents/pp_home könnten durch Pi Presents Updates überschrieben werden.

- es gibt nur eine geringe Kontrolle dieser Dateien durch Pi Presents. Wenn eine Datei modifiziert wurde, sollte pipresents.py in einem Terminalfenster gestartet werden, weil so etwaige Python Fehlermeldungen angesehen werden können.

10.2 Inputs und Outputs einrichten

Kapitel 6.1 beschreibt das Eingabesystem von Pi Presents und zeigt, wie externe Ereignisse in Eingaben umgewandelt werden. Die meisten Systemantworten können mit den in diesem Kapitel beschriebenen Dateien konfiguriert werden.

Beim Editieren der Konfigurationsdateien sind symbolische Namen erforderlich. Es ist ratsam, Namen nicht mit pp- beginnen zu lassen. Das verhindert Kollisionen mit Namen, die Pi Presents benutzt.

10.2.1 GPIO Pins konfigurieren

ACHTUNG: Wenn Sie einen Pin als Ausgang nutzen, wobei der Ausgang mit +3,3 Volt oder Ground kurzgeschlossen wird, könnte das Ihren Pi grillen. Legen Sie zum Schutz vor jeden Ein- und Ausgang einen Serienwiderstand.

Die GPIO-Konfiguration erfolgt in der Datei gpio.cfg. Diese Datei in /pipresents/pp_home/gpio.cfg konfiguriert Pi Presents bei Buttons und PIR, wie in diesem Handbuch beschrieben und in den Beispielen praktisch ausgeführt.

Die Datei gpio.cfg bildet die physischen GPIO Input und Output Pins auf den symbolischen Namen der von Pi Presents gebrauchten Inputs und Outputs ab. In Ihr werden auch die Input-Pins konfiguriert.

Für jeden Pin muss ein Eintrag in der Datei vorhanden sein. Ein Pin mit direction=none wird ignoriert.

Inputs

Jeder Pin kann ein Ereignis auslösen, das seinen symbolischen Namen auf diese vier unterschiedlichen Weisen nutzt:

- rising edge - Ein Ereignis mit dem symbolischen Namen in 'rising-name' wird ausgelöst wenn die Eingabe von 0 zu 1 (0 Volts zu 3,3 Volts) wechselt.

- falling edge - Ein Ereignis mit dem symbolischen Namen in 'falling-name' wird ausgelöst wenn die Eingabe von 1 zu 0 (3,3 Volts zu 0 Volts) wechselt.

- one state – Ein Ereignis, dessen symbolischer Name in ‘one-name’ eingetragen ist, tritt in Intervallen auf, wie sie in 'repeat' eingetragen sind und der Eingabestatus ‘1’ (3,3 Volt) ist. Das Ereignis beginnt nach dem Interval in 'repeat'. Wenn es sofort beginnen soll, tragen sie den gleichen symbolischen Namen beim 'rising edge event' ein.

- zero state – Ein Ereignis, dessen symbolischer Name in ‘zero-name’ eingetragen ist, tritt in Intervallen auf, wie sie in 'repeat' eingetragen sind und der Eingabestatus ‘0’ (0 Volt) ist. Das Ereignis beginnt nach dem Interval in 'repeat'. Wenn es sofort beginnen soll, tragen sie den gleichen symbolischen Namen beim 'rising edge event' ein.

Wenn das Ereignis nicht stattfinden soll, muss der symbolische Name frei bleiben.

Für die Zwecke in diesem Handbuch und die Beispiele ist die gpio.cfg so eingestellt, das normale offene Buttons mit ground (0 Volt) verbunden und eine PIR mit einem normalen Schließkontakt gegen ground funktionieren.

Outputs

Der logische 'ON' Status liefert +3,3 Volt. Der logische 'OFF' Status liefert 0 Volt.

Pi Presents initialisiert GPIO Outputs mit 0 Volt, deshalb ist es sinnvoll, Relais etc. mit positiver Logik zu verwenden.

10.2.2 Klickbereiche einrichten

Die Datei screen.cfg definiert die Bereiche des Bildschirms, die maus- und/oder berührungssensitiv

sein sollen. Die Datei besteht aus mehreren Sektionen mit jeweils einem einmaligen Namen. Der Name ist beliebig, darf in der Datei aber nur einmal vorkommen.

Alle Felder jeder Sektion müssen vorhanden sein. Diese Felder werden wie folgt benutzt:

- name – Der symbolische Name des Klickbereiches. Jeder Befehl im Links Feld des Tracks hat einen symbolischen Namen. Wenn ein Track in der Hyperlinkshow läuft, werden die Klickbereiche des Links Feldes auf dem Bildschirm angezeigt und wenn ein Bereich angeklickt wird, wird über den symbolischen Namen die Art der Eingabe erkannt.

- points – das ist ein Satz von xy-Paaren, die ein Vieleck beschreiben, das den Klickbereich ausmacht. Das Vieleck wird automatisch vervollständigt, deshalb hat ein Viereck 4 (und nicht 5) xy-Paare. Es müssen mindestens 3 Punkt-Paare vorhanden sein. Weitere Einzelheiten und andere Attribute siehe http://effbot.org/tkinterbook/canvas.htm in section create_polygon.

- fill-colour, outline-colour

  Bestimmt das Aussehen des Vielecks. Für Transparenz muss das Feld leer sein.

- text, text-font, text-colour

  Wenn 'text' nicht leer ist, wird der eingetragene Text im Vieleck zentriert.

10.2.3 Tastatur einrichten

Tastaturknöpfe sind symbolischen Namen in der Datei keys.cfg zugeordnet. Die Datei hat mehrere Zeilen im Format:

condition = symbolicname

Die Bedingungen sind in effbot.org/tkinterbook/tkinter-events-and-bindings.htm <Return>, a, und <Shift-Up> Sektionen näher erläutert. Bei Bedingungen und symbolischen Namen sind Groß- und Kleinschreibung zu beachten.

Im Zusammenhang mit diesen Verknüpfungen sind die Buchstaben auf den Tasten (die eine <Key> Bedingung in der Referenz haben) automatisch mit dem symbolischen Namen pp-key-x verbunden

z.B. the 'a' key produces pp-key-a

Die Automatische Verknüpfung einer Taste kann durch eine Zeile wie a=pp-pause überschrieben werden.

Achtung: <any control> = pp-shutdown funktioniert nicht weil pp-shutdown 5 Sekunden gedrückt werden muss. Es taugt nur für gpio. Allerdings kann man den Pi durch Pi Presents mit einem show control Befehl oder der Verknüpfung einer Taste mit dem besonderen symbolischen Namen pp-shutdownnow herunterfahren.

10.2.4 Interne und Laufzeit-Operationen einrichten

Interne Operationen
Die Datei controls.cfg verbindet symbolische Namen mit den internen Operationen von Pi Presents, anders, als andere Konfigurationsdateien, die externe Ereignismeldungen mit den symbolischen Namen verbinden.

Es ist unwahrscheinlich, dass Sie diese Datei verändern wollen, eher könnte es sein, dass Sie das Controls Feld einer Show benutzen wollen, um die Verknüpfung für eine bestimmte Show zu ändern. Der symbolische Name, der mit einer internen Operation verknüpft ist, kann für eine Top-Level-Show und alle ihre Child-/Subshows geändert werden, indem man einen neuen symbolischen Namen für die Funktion im Controls Feld der Top-Level-Show einträgt.
Die Operation für den gesamten Showstapel kann außerdem deaktiviert werden, wenn man den symbolischen Namen mit Null verknüpft. z.B.

In the Controls field of a show:

    mypause pause
    pp-stop null

liefert eine Eingabe, die mypause dazu bringt, statt mit pp-pause alle Tracks dieser Show zu pausieren und verbietet alle Aktionen von pp-stop, wenn die Eingabe die Show, ihre Subshows und Tracks anstößt.

Laufzeitsteuerung
Die Datei controls.cfg kann genutzt werden, um Laufzeitsteuerungsoperationen hinzuzufügen. OMXPlayer und Mplayer haben eine Reihe von Laufzeitsteuerungen, wie hier beschrieben:

https://github.com/popcornmix/omxplayer

und hier

http://www.mplayerhq.hu/DOCS/man/en/mplayer.1.html#INTERACTIVE%20CONTROL

Die Operation hat ein einzelnes Zeichen in der Liste der Operationen, ihm wird omx oder mplay vorangestellt. Mehrfachzeichen-Einträge werden aktuell nicht unterstützt.

Beispiel:

    videovolup = omx-+
    videovoldown = omx--

fügt Lautstärkekontrolle dem OMXPlayer hinzu. Videovolup kann mit einer Taste, einem GPIO-Pin etc. verknüpft werden. Tasten haben von Haus aus eine Wiederholungsfunktion; für GPIO-Pins kann man Wiederholung in der gpio.cfg einrichten.

Für den uzbl browser zum Anzeigen von Webtracks sind die gebräuchlichsten Befehle für die Laufzeitsteuerung hier zu finden:

http://www.uzbl.org/readme.php

Die Befehlszeilen müssen mit uzbl- beginnen.

Beispiel:

largerweb = uzbl-zoom_in

vergrößert die Darstellung des Browsers.

Laufzeit-Operationen sind auch im Controls Feld einer Show geregelt.

10.3 Admin-Nachrichten ändern

Einige für die Nutzer sichtbaren Admin-Nachrichten können nicht mit Profilen beeinflusst werden.

Die Datei resources.cfg ist eine Textdatei, die mit dem Editor Leafpad editiert werden kann. Sie hat einen Bereich für jede Show und jeden Player, in dem Nachrichten für den Zuschauer stehen können. z.B.:

[mediashow]
waiting for user to start a show with progress = manual
m01 = To start the show press 'Play'

waiting for trigger from PIR
m02 =

Es gibt einen Kommentar (#.....) der den Einsatz der Nachricht beschreibt und einen numerischen Eintrag, der sich auf die Nachricht im Python Code bezieht.

Der Text kann nach dem = -Zeichen geändert werden. Wenn nach = leer, erscheint keine Nachricht.

Überprüfen Sie Ihre Einträge in einem Terminalfenster von Pi Presents, so dass Python-Fehlermeldungen angezeigt werden.

Zwei Nachrichten gestatten die Anzeige des Zeitpunktes der nächsten Show. Das erreicht man durch das Einfügen %tt in der Nachrichtenzeile.

11 Plugins

Plugins sind Python-Code-Module mit einer Schnittstellenbeschreibung, die der Erweiterung von Darstellung und Input/Output-Fähigkeiten von Pi Presents dienen:

- Track Plugins gestatten die dynamische Anzeige von Informationen wie Zeit, Wetter und Ticker. Häufig wünscht man sich Informationen von Websites und RSS Feeds.

- Input/Output Plugins gestatten dem Entwickler, durch andere Geräte die vorhandenen GPIO, Tatstaur und Maus zu ersetzen. Möglichweise durch UART/RS232, I2C, DMX over IP oder gewöhnliche andere Geräte, um mehrere Pi Presents zu verbinden. Ein I/O plugin für den Pi gibt es zwar noch nicht, aber die Einrichtung von Eingabeaktionen und symbolischen Namen in der vorliegenden Version ist der Schlüssel dafür.

11.1 Track Plugins

Track Plugins werden von Playern ausgeführt, kurz bevor der zu spielende Track startet. Ihr wesentlicher Zweck ist die Darstellung dynamischer Inhalte. Das kann man auf zwei Wegen erreichen:

- durch Änderung oder Ersetzen des zu spielenden Tracks
- durch Zeigen dynamischer Informationen direkt auf dem Pi Presents Bildschirm.

Track Plugins werden im Verzeichnis /pipresents/pp_home/pp_plugins in .py Dateien gespeichert. Es empfiehlt sich, die Namen der Plugins mit vorangestellten eigenen Initialen zu versehen, um Konflikte mit Python-Modulen zu vermeiden (z.B. time.py würde mit dem Standard-Zeit-Modul kollidieren).

Plugins anwenden

Wenn ein Plugin für einen Track gebraucht wird, muss das 'Plugin Configuration File' Feld eines Players den Namen einer Plugin-Konfigurationsdatei enthalten, z.B. +/media/weather_ny.cfg. relative Pfade sind erlaubt.

Die Plugin-Konfigurationsdatei kann das gleiche Plugin mit unterschiedlichen Parametern aufrufen. Sie muss mindestens enthalten:

```
[plugin]
plugin = pluginname
type = image        # erforderlich, wenn das Plugin in einer Liveshow aufgerufen wird
```

pluginname ist der Name des Python-Moduls, das das Plugin enthält (der Dateiname ohne .py). Plugins werden in /pipresent/pp_home/pp_plugins gespeichert.

Die Konfigurationsdatei kann weitere Parameter angeben, die sich das Plugin via The Dictionary Plugin_Params besorgt:

z.B.

```
[plugin]
plugin = krt_weather
type = image        # erforderlich, wenn das Plugin in einer Liveshow aufgerufen wird
# optional
zip = USNY0996:1:US
place = New York
```

mit mehreren Kofigurationsdateien, die alle das gleiche Wetter-Plugin aufrufen, kann man mit jedem Track einen anderen Wetterbericht anzeigen.

Man kann Plugins in Liveshows nutzen. Dafür müssen Plugins besonders programmiert sein – all meine Beispiele sind es. Um sie in einer Liveshow zu gebrauchen, müssen die Konfigurationsdateien lediglich in das Verzeichnis pp_live_tracks kopiert werden.

Das Beispiel pp_plugin_1p2 zeigt das alles.

Plugins erstellen

Standardmäßig gibt es zwei Plugins von besonderem Interesse für Neu-Entwickler. Sie zeigen die wichtigsten Features und beschreiben die Details der Schnittstelle. Weitere Plugins sind kleinere Varianten dieser zwei für unterschiedliche Tracktypen etc.:

- krt_weather.py verändert die Trackdatei im Trackprofil. Das Plugin benutzt PIL um Informationen von einer Wetterseite abzugreifen, speichert das modifizierte Bild und gibt den Pfad weiter, so dass es der Imageplayer darstellen kann. Dieses Beispiel dokumentiert bestens das API.

Das Python Imaging Library (PIL) Handbuch ist hier: http://effbot.org/imagingbook/pil-index.htm.

Tkinter canvas Operationen sind hier dargestellt http://effbot.org/tkinterbook/canvas.htm

Die Funktionsschnittstelle verändert sich relativ wenig. Das hängt vom Tracktyp ab:

- image – das Plugin erwartet den Pfad zu einer Datei, die ein Bild enthält (z.B. picture.jpg). Eine Bilddatei kann wiedergegeben werden. Wenn sie wiedergegeben wird, muss sie für die Anzeige mit PIL und Tkinter geeignet sein. Das Plugin kann auch direkt in Canvas schreiben.

- audio – das Plugin wird mit einer Audiodatei „gefüttert“. Der Audiotrack kann für das Abspielen im MPlayer ausgegeben werden. Leer ist erlaubt, dann wird kein Audio gespielt. Das Plugin kann auch direkt in Canvas schreiben.

- video – das Plugin wird mit einer Videodatei versorgt. Der Videotrack kann für das Abspielen im OMXPlayer ausgegeben werden. Leer ist erlaubt, lässt den Videoplayer aber sofort enden. Das Plugin kann auch direkt in Canvas schreiben, aber achten Sie darauf, dass das Video über dem oberen Rand des Textes erscheint. Richten Sie die Position mit video_window ein.

- web – das Plugin wird mit einer Datei versorgt, die HTML-Code enthält (was immer auch uzbl rendern kann). Eine HTML-Datei kann für das Abspielen in uzbl ausgegeben werden. Das Plugin kann auch direkt in Canvas schreiben, aber achten Sie darauf, dass der Browser über dem oberen Textrand erscheint. Richten Sie seine Position mit Web Window ein.

- message – das Plugin wird mit dem ursprünglichen Text versorgt, der geändert und zurück gegeben wird. Das Plugin kann auch direkt in Canvas schreiben.

11.2 Input/Output Plugins

TBD

12 Anforderunge an die Hardware

Pi Presents läuft sowohl auf Revision 1 als auch Revision2 des Raspberry Pi. Die GPIO-Pins wurden so ausgewählt, dass die Version irrelevant ist. 256-MB-Pi können mit Bildgrößen bis zu einem Megapixel umgehen.

Das Aufsplitten des Arbeitsspeichers hängt vom OMXPlayer ab und auf 256-MB-Maschinen sollte das Minimum, das der OMXPlayer benötigt, so sein, wie das Maximum Arbeitsspeicher für die Verarbeitung von Bildern. Der OMXPlayer spielt einige Videos unter 64MB Arbeitsspeicher ab, andere Videos benötigen 128 MB, besonders wenn Untertitel dabei sind.

Die Bildanzeige ist langsam. Ein 1-Megapixel-Bild benötigt zum Aufbau etliche Sekunden. Ein 10-Megapixel-Bild benötigte im Test 10 Sekunden und brachte einen 256-MB-Pi zum Absturz. Bildauflösungen, größer als die Bildschirmauflösung, bringen keine bessere Bildqualität und benötigen auch auf 512-MB-Maschinen lange Ladezeiten. Ich benutze den brillianten Faststone Photo Resizer http://www.faststone.org/FSResizerDetail.htm um die Bildgrößen auf einem Windowsrechner zu verringern.

Für den Sound benutzen Sie HDMI oder den 3,5 Klinkenausgang für Audio. Verstärkung und Lautstärkekontrolle könnten mit externer Hardware erfolgen und so die Anwendung ergänzen.

13 Pi Presents updaten

Zur Sicherheit fertigen Sie eine Kopie des Pi Presents Verzeichnisses und Ihrer Daten an, bevor Sie ein Update starten.

Laden Sie Pi Presents von Github herunter und installieren Sie es, wie in der Datei README.md beschrieben. Daten in /home/pi/pp_home werden nicht angerührt. Speichern Sie Ihre Daten nicht in pipresents/pp_home Verzeichnis, weil sie dort überschrieben werden könnten.

Weil Pi Presents sich weiter entwickelt, werden Felder zu den Profildefinitionen hinzugefügt, andere gelöscht. Um das sicherzustellen müssen Pi Presents, der Editor und die Profile die gleiche Version tragen. Pi Presents bemerkt es, wenn ein Profil mit falscher Versionsnummer benutzt wird. Um das zu korrigieren, öffnen Sie dieses Profil im Editor, der das Profil und seine Felder automatisch updated. Es gibt einige wenige Update-Schritte, die nicht automatisiert werden können. Lesen Sie die Release Notes, um mehr darüber zu erfahren.

Wenn Sie viele Profile im gleichen /pp_home updaten müssen, können Sie 'tools>update all' aus dem Editormenü benutzen, um das Update elegant zu realisieren.

Die Profile in pipresents-examples werden kompatibel zur jeweils letzten Pi Presents Version gehalten. Achtung: Sie zu installieren könnte ihre älteren Profile überschreiben.

Wenn Sie Pi Presents updaten, lesen Sie bitte die Versionshinweise. Sie müssen möglicherweise die Konfigurationsdateien updaten und einige wenige Aufgaben erledigen, die nicht automatisch ablaufen.

14 Fehlerreports und Programmwünsche

Bitte benutzen Sie Github Issues Tab https://github.com/KenT2/pipresents/issues oder den Pi Presents thread http://www.raspberrypi.org/phpBB3/viewtopic.php?f=38&t=29397 im Raspberry Pi Forum um Fehler zu melden.

Ich bin versessen darauf, Pi Presents zu verbessern. Ihre Erfahrungen im Echtbetrieb und Ihre Wünsche an das Programm sind für mich von unschätzbarem Wert, weil sie mir beim Feintuning der aktuellen Funktionalität im Kleinen und bei größeren Verbesserungen helfen.

15 Lösungen und bekannte Probleme

Wenn nicht Vollbild eingestellt ist, ist das Pi Presents Fenster entweder zu klein oder zu groß

Anstehende Einführung eines Options-Editors. Ändern Sie folgende Zeilen in pipresents.py (ungefähr Zeile 40).

```
self.nonfull_window_width = 0.6 # proportion of width
self.nonfull_window_height = 0.6 # proportion of height
self.nonfull_window_x = 0 # position of top left corner.py
self.nonfull_window_y = 0   # position of top left corner
```

Mit sudo, autostart, oder Start mit einer Desktopverknüpfung wird das Profil nicht gefunden
In diesen Fällen ist es das Beste, den absoluten Pfad von Pi Presents und Daten-Heimatverzeichnis in die Kommandos aufzunehmen z.B.

```
sudo python /home/pi/pipresents.py -o /home/pi -p myprofile
```

Pi Presents hängt im Vollbildschirm fest, wie kann ich ein da herauskommen?
Das liegt daran, dass Python eine Ausnahme berichtet, die auf inkorrekte Konfigurationsdaten zurückzuführen ist.

Um das zu vermeiden, benutzen Sie Validation im Editor und testen die Show, nicht im Vollbildschirm aber in einem Terminalfenster, dann geht CTRL-C, wenn CTRL-BREAK nicht funktioniert. Wenn das fehlschlägt, versuchen Sie das Terminalfenster zu schließen und wenn nötig, neu zu booten. Eine Stromunterbrechung könnte das System korrumpieren, so wäre eine saubere Lösung, von einem anderen Rechner mit SSH auf den Pi zuzugreifen, top (top- upi) zu starten und den Pythonprozess mit kxxxx Befehl abzubrechen. Möglich, dass auch der OMXPplayer Prozess abgebrochen werden muss.

Mein Video/Audiotrack läuft nicht in Pi Presents
Versuchen Sie beide mit OMXPlayer oder MPlayer von der Befehlszeile abzuspielen.

Unclutter versteckt den Cursor nicht durchgehend
Azizar berichtete das er /etc/default/unclutter, geändert hat und folgendes zugefügt wurde:

```
-grab
-noevents
```

Ich habe ein Input- oder Outputgerät eingerichtet, aber es funktioniert nicht mit Pi Presents

Es gibt zwei einzeln funktionierende GPIO-Testprogramme im pipresents Verzeichnis 'input_test.py' und 'output_test.py'. Damit können Sie Ihre I/O testen bevor Sie Pi Presents tadeln!

Erlaubnis in pp_log.log zu schreiben wird verweigert
Die Logdatei pp_log.log wurde mit den Rechten des aktuellen Nutzers erstellt. Wenn sudo im Befehl zur Dateierstellung benutzt wurde, hat Root die Rechte. Löschen Sie als Benutzer Root die Datei und legen dann als Benutzer Pi an oder führen pipresents.py ohne sudo aus.

Manchmal stürzt Pi Presents ab, wenn zwei Audiotracks abgespielt werden
Ich habe Folgendes herausgefunden: wenn zwei Audiotracks mit dem MPlayer gespielt werden, wobei einer über HDMI und der andere lokal ausgegeben wird, wechselt der Tonausgangskanal und es lässt den Mplayer abstürzen.

Pi Presents stürzt ab, wenn ein Video und ein Audiotrack gleichzeitig gespielt werden
Wenn der vom MPlayer gespielte Audiotrack endet, stürzt der Raspberry Pi ab. Ich werde herausfinden, ob das in der neuesten Firmware passiert.

Wenn ich den Editor in Windows benutze kommt die Meldung: "Fehler beim Speichern der Medialiste – noch einmal versuchen."
Unter Windows wird die Medialiste zunächst nicht periodisch abgespeichert. Wenn Sie es erneut probieren, könnte es Erfolg haben. Ideen, warum dies vorkommt, sind willkommen. Der Code liegt in der pp_medialist.py bei ungefähr Zeile 266.

Pi Presents stürzt ab, wenn Videos ein paar Stunden lang abgespielt werden
Es scheint ein Timing-Problem im Zusammenspiel oder sogar mit dem OMXPlayer selbst zu geben. Nach einem Absturz ist ein Neustart des Pi angebracht, weil der OMXPlayer sich in einem ungewissen Zustand zu befinden scheint.